

Interdisciplinary Mechatronics

Interdisciplinary Mechatronics

Engineering Science and Research Development

Edited by
Maki K. Habib
J. Paulo Davim

ISTE

WILEY

First published 2013 in Great Britain and the United States by ISTE Ltd and John Wiley & Sons, Inc.

Apart from any fair dealing for the purposes of research or private study, or criticism or review, as permitted under the Copyright, Designs and Patents Act 1988, this publication may only be reproduced, stored or transmitted, in any form or by any means, with the prior permission in writing of the publishers, or in the case of reprographic reproduction in accordance with the terms and licenses issued by the CLA. Enquiries concerning reproduction outside these terms should be sent to the publishers at the undermentioned address:

ISTE Ltd
27-37 St George's Road
London SW19 4EU
UK

www.iste.co.uk

John Wiley & Sons, Inc.
111 River Street
Hoboken, NJ 07030
USA

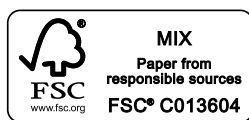
www.wiley.com

© ISTE Ltd 2013

The rights of Maki K. Habib and J. Paulo Davim to be identified as the author of this work have been asserted by them in accordance with the Copyright, Designs and Patents Act 1988.

Library of Congress Control Number: 2013930461

British Library Cataloguing-in-Publication Data
A CIP record for this book is available from the British Library
ISBN: 978-1-84821-418-7



Printed and bound in Great Britain by CPI Group (UK) Ltd., Croydon, Surrey CR0 4YY

Table of Contents

Preface	xvii
Chapter 1. Interdisciplinary Mechatronics Engineering Science and the Evolution of Human Friendly and Adaptive Mechatronics	1
Maki K. HABIB	
1.1. Introduction.	2
1.2. Synergetic thinking, learning and innovation in mechatronics design	9
1.3. Human adaptive and friendly mechatronics.	11
1.4. Conclusions.	14
1.5. Bibliography	15
Chapter 2. Micro-Nanomechatronics for Biological Cell Analysis and Assembly	19
Toshio FUKUDA, Masahiro NAKAJIMA, Masaru TAKEUCHI, Tao YUE and Hirotaka TAJIMA	
2.1. Introduction of micro-nanomechatronics on biomedical fields	19
2.2. Configuration of micro-nanomechatronics	21
2.3. Micro-nanomechatronics for single cell analysis.	25
2.4. Semi-closed microchip for single cell analysis	28
2.5. Biological cell assembly using photo-linkable resin based on the single cell analysis techniques	30
2.6. Conclusion	33
2.7. Acknowledgments.	34
2.8. Bibliography	34

Chapter 3. Biologically Inspired CPG-Based Locomotion Control System of a Biped Robot Using Nonlinear Oscillators with Phase Resetting	37
Shinya AOI	
3.1. Introduction.	37
3.2. Locomotion control system using nonlinear oscillators	38
3.2.1. CPG-based locomotion control system	38
3.2.2. Rhythm generator model using nonlinear oscillators	39
3.2.3. Pattern formation model to determine the global parameters of limb kinematics	40
3.2.4. Phase regulation based on interlimb and intralimb coordination	40
3.2.5. Sensory regulation based on phase resetting	40
3.3. Stability analysis using a simple biped robot model	41
3.3.1. Compass model	41
3.3.2. Locomotion control system	43
3.3.3. Periodic solution of the walking motion	45
3.3.4. Stability analysis	49
3.4. Experiment using biped robots	58
3.4.1. Locomotion control system	58
3.4.2. Experimental results	61
3.5. Conclusion	64
3.6. Acknowledgments	65
3.7. Bibliography	65
Chapter 4. Modeling a Human's Learning Processes toward Continuous Learning Support System	69
Tomohiro YAMAGUCHI, Kouki TAKEMORI and Keiki TAKADAMA	
4.1. Introduction.	70
4.1.1. The need for continuous learning in mechatronics engineering	70
4.1.2. Learning support systems for a human	70
4.1.3. Modeling a learning process to achieve continuous learning	71
4.1.4. How to keep supplying new goals to achieve continuous learning	73
4.1.5. The concept to formalize the continuous learning by a maze model	75
4.2. Designing the continuous learning by a maze model	76
4.2.1. A learning environment by a maze model with invisible goals	76
4.2.2. The maze sweeping task that involves multiple goals	78
4.2.3. Designing the thinking level	81
4.3. The layout design of mazes for the continuous learning task.	82
4.3.1. Overview of the continuous learning support system	82
4.3.2. The layout design of mazes on the thinking level space	83

4.4. Experiment	85
4.4.1. Experimental setup	85
4.4.2. Experimental results	87
4.5. Discussions	88
4.5.1. The role of motivations to drive the continuous learning	88
4.5.2. Why is it important to collect all solutions for continuous learning?	89
4.5.3. Toward an application of a maze model and invisible goals.	90
4.6. Conclusions.	92
4.7. Acknowledgments.	93
4.8. Bibliography	93
 Chapter 5. PWM Waveform Generation Using Pulse-Type Hardware Neural Networks	 95
Ken SAITO, Minami TAKATO, Yoshifumi SEKINE and Fumio UCHIKOBA	
5.1. Introduction.	96
5.2. PWM servo motor	97
5.3. Pulse-type hardware neuron model	99
5.3.1. Basic cell body model	99
5.3.2. Cell body model for PWM controlling system.	101
5.3.3. Synaptic model	103
5.3.4. Axon model	104
5.4. Pulse-type hardware neural networks	104
5.5. Measurements of constructed discrete circuit.	108
5.6. Conclusion	109
5.7. Acknowledgments.	109
5.8. Bibliography	110
 Chapter 6. Parallel Wrists: Limb Types, Singularities and New Perspectives.	 113
Raffaele DI GREGORIO	
6.1. Limb architectures and mobility analysis	113
6.1.1. Use of the screw theory	115
6.1.2. Use of the group theory	120
6.1.3. Other approaches	122
6.1.4. Conclusion.	124
6.2. Singularities and performance indices	124
6.2.1. Singularity analysis of PWs.	126
6.2.2. Kinetostatic performances.	131
6.2.3. Conclusion.	138
6.3. New perspectives	139
6.4. Bibliography	142

Chapter 7. A Robot-Assisted Rehabilitation System – RehabRoby	145
Duygun EROL BARKANA and Fatih ÖZKUL	
7.1. Introduction.	145
7.2. Background.	146
7.2.1. Robot-assisted rehabilitation systems for upper extremities.	146
7.2.2. Controllers of robot-assisted rehabilitation systems for upper extremities.	148
7.3. Control architecture	149
7.4. RehabRoby	150
7.5. Controllers of RehabRoby	155
7.5.1. Low-level controller	156
7.5.2. High-level controller	157
7.6. Concluding remarks.	158
7.7. Acknowledgments.	159
7.8. Bibliography	159
 Chapter 8. MIMO Actuator Force Control of a Parallel Robot for Ankle Rehabilitation.	 163
Andrew MCDAID, Yun HO TSOI and Shengquan XIE	
8.1. Introduction.	163
8.1.1. Rehabilitation robots	164
8.1.2. Ankle sprain rehabilitation	165
8.2. Ankle rehabilitation robot	167
8.2.1. Design requirements	168
8.2.2. Analysis for four-link parallel mechanism	171
8.2.3. System description	175
8.3. Actuator force control.	176
8.3.1. Actuator force control by decoupling of inertia matrix.	177
8.3.2. Higher order dynamic model of actuator–sensor–environment system	182
8.3.3. State space model of the linearized actuator–sensor–environment system	187
8.3.4. Proposed actuator force controller	195
8.4. Experimental results.	198
8.4.1. Stability experiment	198
8.4.2. Experiments for performance evaluation	200
8.5. Concluding remarks.	204
8.6. Bibliography	205

Chapter 9. Performance Evaluation of a Probe Climber for Maintaining Wire Rope	209
Akihisa TABATA, Emiko HARA and Yoshio AOKI	
9.1. Introduction.	209
9.2. Optimize friction drive conditions using a prototype probe climber	210
9.3. Impact of different surface friction materials for friction pulley made on elevation performance	213
9.4. Damage detection test of elevator wire rope	216
9.5. Damage detection through signal processing	218
9.6. Integrity evaluation of wire rope through MFL strength	219
9.7. Damage detection of wire rope using neural networks	224
9.8. Conclusion	224
9.9. Bibliography	225
 Chapter 10. Fundamentals on the Use of Shape Memory Alloys in Soft Robotics.	227
Matteo CIANCHETTI	
10.1. Introduction	228
10.2. Shape memory effect and superelastic effect	230
10.3. SMA thermomechanical behavior.	231
10.4. SMA constitutive models.	234
10.5. Hints on SMA thermomechanical testing	235
10.6. Design principles.	237
10.6.1. Geometrical choice	237
10.6.2. One-way or two-way memory effect.	240
10.6.3. Restoring force	240
10.6.4. Anchor points	241
10.7. Fabrication methods	243
10.8. Activation methods and control design.	244
10.8.1. On-off activation in open-loop control	246
10.8.2. Modulated activation in closed-loop control	247
10.9. Applications in Soft Robotics	248
10.10. Conclusions	251
10.11. Bibliography.	252
 Chapter 11. Tuned Modified Transpose Jacobian Control of Robotic Systems	255
S. A. A. MOOSAVIAN and M. KARIMI	
11.1. Introduction	256
11.2. TMTJ control law	257
11.2.1. Feedback linearization approach	257

11.2.2. Transpose Jacobian algorithm	259
11.2.3. Modified transpose Jacobian algorithm	259
11.2.4. Tuned modified transpose Jacobian algorithm	261
11.3. Obtained results and discussions	265
11.3.1. Fixed base manipulator.	265
11.3.2. Mobile base manipulator.	269
11.4. Conclusions	272
11.5. Bibliography	273
Chapter 12. Derivative-Free Nonlinear Kalman Filtering for PMSG Sensorless Control	277
Gerasimos RIGATOS, Pierluigi SIANO and Nikolaos ZERVOS	
12.1. Introduction	277
12.2. Dynamic model of the permanent magnet synchronous generator. . .	279
12.3. Lie algebra-based design of nonlinear state estimators	282
12.3.1. Relative degree for nonlinear systems	282
12.3.2. Nonlinear observer design for exactly linearizable systems	283
12.3.3. Linearization of PMSG dynamics using Lie Algebra	287
12.4. Differential flatness for nonlinear dynamical systems	288
12.4.1. Definition of differentially flat systems	288
12.4.2. Classes of differentially flat systems.	291
12.4.3. Differential flatness and transformation into the canonical form	292
12.5. Differential flatness of the PMSG.	293
12.6. Robust state estimation-based control of the PMSG	296
12.6.1. Unknown input observers	296
12.6.2. Perturbation observer.	296
12.6.3. Extended state observer	297
12.7. Estimation of PMSG disturbance input with Kalman filtering	298
12.7.1. State estimation with the derivative-free nonlinear Kalman filter	298
12.7.2. Kalman filter-based estimation of disturbances	299
12.8. Simulation experiments	302
12.9. Conclusions	307
12.10. Bibliography.	308
Chapter 13. Construction and Control of Parallel Robots	313
Moharam HABIBNEJAD KORAYEM, Soleiman MANTEGHI and Hami TOURAJIZADEH	
13.1. Introduction	313
13.2. A parallel robot mechanism	315

13.2.1. Mechanical structure	316
13.2.2. Modeling and Formulation	319
13.3. Actuators	324
13.3.1. Motor	324
13.3.2. Cables	327
13.4. Sensors.	328
13.4.1. Motor speed sensor	328
13.4.2. Force sensor	330
13.4.3. Position sensor.	334
13.5. Data transfer protocol	342
13.6. Graphical user interface (GUI).	347
13.6.1. Simulator and virtual laboratory	347
13.6.2. Hardware control	352
13.7. Result and verifications	357
13.8. Conclusion	362
13.9. Bibliography	364
Chapter 14. A Localization System for Mobile Robot Using Scanning Laser and Ultrasonic Measurement	369
Kai LIU, Hongbo LI and Zengqi SUN	
14.1. Introduction	369
14.2. System configuration	371
14.3. Implementation	373
14.3.1. Dead reckoning system.	373
14.3.2. Scanning laser and ultrasonic positioning system	375
14.4. Experimental results	377
14.5. Conclusion	382
14.6. Acknowledgments	383
14.7. Bibliography	383
Chapter 15. Building of Open-Structure Wheel-Based Mobile Robotic Platform.	385
Aleksandar RODIĆ and Ivan STOJKOVIĆ	
15.1. Introduction	385
15.2. State of the art	386
15.3. Configuring of the experimental system	389
15.3.1. Wheel-based mobile robot.	390
15.3.2. System for localization and obstacle detection	390
15.3.3. Architecture of the on-board controller	391
15.4. Modeling and simulation of the system	394
15.4.1. Kinematical model	395
15.4.2. Model of robot dynamics	397

15.5. Motion planning and control	403
15.5.1. Motion planning	403
15.5.2. Motion control	406
15.6. Simulation and experimental testing	409
15.6.1. Case studies	411
15.7. Concluding remarks	416
15.8. Acknowledgments	417
15.9. Bibliography	417
15.10. Appendix.	421
 Chapter 16. Design and Physical Implementation of Holonomous Mobile Robot – Holbos	 423
Jasmin VELAGIC, Admir KAKNJO, Faruk DAUTOVIC, Muhidin HUJDUR and Nedim OSMIC	
16.1. Introduction	423
16.2. Locomotion of holonomous mobile robot	424
16.2.1. Kinematic model of robot	426
16.3. Mechanical design	430
16.4. Electrical design	431
16.4.1. Hardware components and software implementation	431
16.5. Results	444
16.6. Conclusion	447
16.7. Bibliography	448
 Chapter 17. Advanced Artificial Vision and Mobile Devices for New Applications in Learning, Entertainment and Cultural Heritage Domains	 451
Gian Luca FORESTI, Niki MARTINEL, Christian MICHELONI and MARCO VERNIER	
17.1. Introduction	451
17.2. Chapter contributions	455
17.3. Mobile devices for education purposes.	456
17.3.1. Mobile learning for education.	457
17.3.2. Mobile learning for museums and galleries.	459
17.3.3. Mobile learning for workplace, professional development and training	460
17.4. Image processing supports HCI in museum application	461
17.4.1. System description	463
17.4.2. Boundary detector.	463
17.4.3. Signature computation	464
17.4.4. Signature matching	466

17.4.5. Homography estimation	467
17.4.6. Human device interface	467
17.4.7. Experimental results	468
17.5. Back to the Future: a 3D image gallery.	471
17.5.1. System description	473
17.5.2. Testing and evaluations	475
17.6. Conclusions and future works	477
17.7. Bibliography	477
 Chapter 18. Application of Stereo Vision and ARM Processor for Motion Control	 483
Moharam HABIBNEJAD KORAYEM, Michal IRANI and Saeed RAFEE NEKOO	
18.1. Introduction	483
18.2. Stereo vision	486
18.3. Triangulation	487
18.4. End-effector orientation.	490
18.5. Experimental setup and results.	492
18.6. Summary	497
18.7. Bibliography	498
 Chapter 19. Mechatronics as Science and Engineering – or Both	 501
Balan PILLAI and Vesa SALMINEN	
19.1. Introduction	501
19.2. Theories and methods of design, planning and manufacturing	504
19.3. Complexity versus complicatedness	506
19.4. Benefits of fast product developments	513
19.5. Nature of product development process	516
19.6. Planning the timetable of a product design project	518
19.7. Designing the product concept.	520
19.8. Enhancing conceptual design	520
19.9. Interaction between the parts of the machine	523
19.10. Effect of the strength of interaction between product parts and development speed	524
19.11. Definition of product and service	527
19.12. The case studies.	529
19.13. Networking systems and learning mechanism.	531
19.14. Model-based methodology: an implemented case.	536
19.15. Conclusions	540
19.16. Bibliography.	541

Chapter 20. A Mechatronic Platform for Robotic Educational Activities	543
Ioannis KOSTAVELIS, Evangelos BOUKAS, Lazaros NALPANTIDIS and Antonios GASTERATOS	
20.1. Introduction	543
20.2. System overview	545
20.2.1. Architectural design	546
20.2.2. Subsystem equipment	547
20.2.3. Robot montage	553
20.3. Educational activities	554
20.3.1. 3D reconstruction	554
20.3.2. Visual odometry	556
20.3.3. Visual SLAM	558
20.3.4. Human tracking using RGB-D sensor	558
20.3.5. Hardware acceleration	559
20.3.6. Sensor fusion algorithms	560
20.3.7. User interfaces	560
20.4. Experiences from educational activities	561
20.5. Conclusions	565
20.6. Acknowledgments	565
20.7. Bibliography	566
Chapter 21. The Importance of Practical Activities in the Formation of Mechatronic Engineers	569
João Carlos M. CARVALHO and Vera Lúcia D.S. FRANCO	
21.1. Introduction	569
21.2. Curricular and extracurricular practical activities	575
21.2.1. Practical laboratory activities	576
21.2.2. Monitoring	577
21.2.3. Research initiation activities	577
21.2.4. Participation in Junior Companies	577
21.2.5. Academic mobility	578
21.2.6. Participation in competition teams	579
21.2.7. Participation in student directories	579
21.2.8. Participation and organization of events	580
21.3. Undergraduate course of Mechatronics Engineering at the Federal University of Uberlândia/Brazil	580
21.3.1. Practices activities of laboratory	581
21.3.2. Monitoring	583
21.3.3. Research initiation activities	583
21.3.4. Participation in Junior Companies	584
21.3.5. Academic mobility	584

21.3.6. Participation in competition teams	585
21.3.7. Participation in student directories	587
21.3.8. Participation and organization of events.	587
21.3.9. Other activities.	587
21.4. Discussions	588
21.5. Conclusions	590
21.6. Bibliography	591
List of Authors	593
Index	599

Preface

The term “mechatronics” was first introduced in 1969 and since then it has evolved very fast and has been used in different dimensions and applications: as practices, as technologies, as a philosophy and as an educational discipline. Mechatronics represents a unifying interdisciplinary and intelligent engineering science paradigm. It is a holistic, concurrent and interdisciplinary engineering science discipline that brings out novel possibilities of synergizing and fusing different disciplines. Mechatronics concentrates on achieving optimum functional synergy from the earliest conceptual stages of the design process. The adoption of such a synergized inter- or transdisciplinary approach to engineering design implies a greater understanding of the design process. Mechatronics systems, processes and products enable the implementation of a scalable architecture in which functional elements are equipped with local control, sensing and diagnostics, intelligence and communication features. The design of mechatronical systems must ensure an effective transfer of complexity between individual technologies, both for individual elements making the system and for the system as a whole.

This book contains materials covering the state-of-the-art research and development within the interdisciplinary mechatronics engineering science and its educational evolution. This book includes 21 chapters, divided into five groups.

The first group covered by Chapter 1, presents mechatronics as an interdisciplinary engineering science and the evolution of human adaptive and friendly mechatronics (HAFM) while looking into the evolution of its educational programs and the necessity to develop potential programs fulfilling the need of different educational levels.

The second group includes chapters 2–5. It introduces micro-nanomechatronics and biological cell assembly, biological inspiration and the use of a central pattern generator (CPG)-based locomotion, hardware neural networks, etc. In addition, it describes the modeling of a human’s learning processes towards achieving a continuous learning support system.

The third group covers chapters 6–10. It presents a range of topics focusing on biomechanics, rehabilitation and the development and evaluation of mechatronical robots of different structures and mechanisms supporting different applications. In addition, Chapter 10 presents the fundamentals on the use of shape memory alloys in soft robotics.

The fourth group consists of chapters 11–18. These chapters discuss different topics related to the control issues of different robotic mechanisms and physical structures, nonlinear Kalman filtering and the integration of sensors with motion. In addition, there are a few chapters on mobile robot design, implementation, navigation, localization and control.

The fifth group includes chapters 19–21. It presents mechatronics, the educational activities, practices and the needs for the formation of mechatronics engineers.

Interdisciplinary mechatronics engineering science helps to prepare engineers and scientists who are able to develop innovative and intelligent ideas for autonomous and smart interdisciplinary products and systems to meet today’s most pressing challenges.

This book is aimed at mechatronics students, engineers and scientists, and also at graduate students, robotics engineers and researchers, and practicing engineers, who wish to enhance and broaden their knowledge and expertise on the fundamentals, practices, technologies, applications and the evolution of mechatronics as an interdisciplinary engineering science.

Maki K. HABIB
J. Paulo DAVIM
March 2013

Chapter 1

Interdisciplinary Mechatronics Engineering Science and the Evolution of Human Friendly and Adaptive Mechatronics

Mechatronics represents a unifying interdisciplinary and intelligent engineering science paradigm [HAB 07]. It describes the synergy of education in science and technology, knowledge, learning, thinking, a way of working, practices and professional skills. Interdisciplinary mechatronics is regarded as a philosophy that supports new ways of thinking, innovations and novel possibilities and new design methodologies that aim to achieve optimum functional synergy. Interdisciplinary mechatronics is supported by a strong engineering science foundation beyond the traditional disciplinary boundaries and features open-ending learning chains with ability to fuse new knowledge and experiences through lifelong learning process. Mechatronics engineering is about mastering a multitude of disciplines, technologies and their interaction, whereas mechatronic science is about invention and the development of new theories, models, concepts and tools in response to the needs evolving from interacting scientific disciplines. This philosophy has led to the development of new products, processes and systems that exhibit quality performance, such as reliability, precision, smartness, flexibility,

2 Interdisciplinary Mechatronics

adaptability, robustness and economical features. Human adaptive and friendly mechatronics (HAFM) represents the new era of mechatronics [HAB 07, HAB 08a, HAB 08b, HAB 12].

The evolution of interdisciplinary mechatronics and the growth of interest in this field triggered a need to have a quality of engineers whose education and training enables them to operate in an interdisciplinary environment. This demanded further the development of an efficient curriculum structure of a mechatronics engineering degree program designed to provide in-depth interdisciplinary knowledge and to focus on individual and team-based learning through a problem and project-based learning approach to fulfill the challenges of modern technology and innovation [HAB 08a].

1.1. Introduction

The new frontiers of engineering have a significant impact on engineering practices, and the formal boundaries among traditional engineering disciplines have become fuzzy following the advent of reliable integrated circuits, sensor technologies, computers and software engineering [HAB 06]. To meet the challenges of the new frontiers of engineering sciences and the needs of the 21st Century, it is important to educate engineers to think across multiple disciplines. Many of the new frontiers involve a high degree of interdisciplinarity and require a strong engineering science foundation.

Mechatronics has come to mean the synergy and the interdisciplinarity of engineering science that successfully fuses (but not limited to) mechanics, electrical, electronics, informatics and intelligent systems, control systems and advance modeling, optics, systems engineering, artificial intelligence, intelligent computer control, precision engineering, virtualization and virtual environments into a unified framework that enhances the design of products and manufacturing of processes. The synergy in mechatronics engineering creative design and development enables a higher level of productivity, precise performance, smart and high functionality, precision, robustness, power efficiency, application flexibility and modularity, improved quality and reliability, intelligence, maintainability, efficiency, energy and environmental consideration, better spatial integration of subsystems (embodied systems), miniaturization, and embed lifecycle design considerations. In addition, it contributes to shorten

development time, to deal flexibly with complexity and to adopt cost-effective approaches for a wide range of intelligent products and applications. It is important to note that the mechatronics concept is not just about achieving technological and knowledge synergy but it also involves aspects of organization, training, management (projects, products, market demands and administration) and sustainable development. Mechatronics systems, processes and products enable the implementation of a scalable architecture in which functional elements are equipped with local control, diagnostics and communication features. The design of mechatronical systems must ensure an effective transfer of complexity between individual technologies, both for individual elements within the system and for the system as a whole. The adoption of such a synergized, inter- or transdisciplinary approach to engineering design implies a greater understanding of the design process [HAB 07].

The term “mechatronics” was first coined by Tetsuro Mori, CEO and president of Seibu Electric and Machinery Co Ltd. in 1969 when he worked for the Yaskawa Electric Corporation in Kitakyushu, Japan [KYU 96]. “Mechatronics” was used in his proposal to describe a new technology focus to produce new electromechanical machine tools that unites mechanisms and electronics while supported by semiconductor power devices and CPUs. “Mechatronics” has, since then, taken on a wider meaning than the traditional term of electromechanics. Figure 1.1 shows the evolution of mechatronics as a technology, practices, a concept and an engineering science discipline. In addition, mechatronics is now widely used to describe a philosophy in engineering and not just the technology itself [HAB 07].

During the early 1970s, mechatronics was simple and focused mostly on servo technology used in products with electronically controlled mechanisms, such as automatic door openers, vending machines and auto-focus cameras. At this stage, Mechatronics products were simple in implementation and encompassed the early use of advanced control methods, while the applied technologies were developed individually and independently of each other. With the development of ever more sophisticated technologies, the technical know-how has become a primary competitive factor, and efficient technology management were extremely important to keep track of the fast development in a large number of fields. The introduction of microprocessors in the mid-1970s has produced an explosion in the development of new products that incorporate the technologies associated with mechanical engineering, electronics and information processing. Hence, there has been an increasing need for a new approach to engineering. During the late 1960s and 1970s, the

4 Interdisciplinary Mechatronics

term “system engineering” came into widespread usage and the design methods used in mechatronics are derived substantially from the techniques involved by system engineers [TRE 03]. System engineers were the first to deal with the complex issues raised by interactions between software, mechanical engineering and electronics.

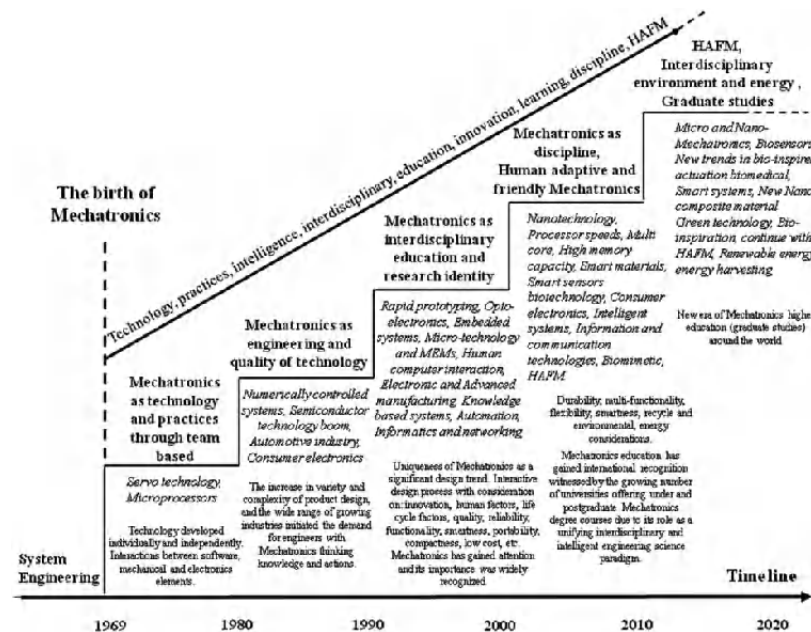


Figure 1.1. Evolution of mechatronics as an interdisciplinary engineering science field. Adopted and modified from [HAB 07]

Mechatronics emerged in Japan to describe the Japanese practices of using fully integrated teams of product designers with other members that represent engineers with traditional engineering backgrounds, manufacturing staff, purchasing and marketing personnel, etc. Such approach facilitates the development of a practical base approach supported by traditional theory to cover the key areas of practical design. In addition, mechatronics at this stage reflects the way people work to design and implement high performance, simple and reliable solutions. Nevertheless, most engineers working in teams rely on people outside their circle for implementing the technical solutions.

This demands spending much of their time on communication and the coordination of activities, and organizing technical solutions rather than implementing them. Mechatronics evolves as a design philosophy and a methodology used to optimize the design process, raises synergy and provides catalytic effect for discovering new and simpler solutions to traditionally complex problems. Such methodology is a collection of practices, procedures and rules used by those who work in a particular branch of knowledge or discipline.

In the 1980s, mechatronics came to mean engineering and technology. The advances in digital electronics have enabled the possibility to invent, create and improve systems that rely on mechanical components to perform their intended actions. A synergistic integration of different technologies started taking place, a notable example being in optoelectronics. During this stage, the codesign concept of hardware/software has been developed and used. The development of mechatronics technology was initially driven by the explosive trend in automation within the automobile industry and the increased use of electronic content in vehicles and the enhanced features of the control systems via software. Other contributing industries include industrial machinery and numerically controlled systems, product integration and manufacturing in consumer electronics and the semiconductor industry. During the second half of the 1980s, the mechatronics quality product lifecycle started to be compressed dramatically in which new technology has to be developed, manufactured and introduced to the marketplace, in a cost effective manner and quickly ahead of the competition, as early market entry for the product provides a critical competitive edge.

The enormous increases in variety and complexity of product design and the wide range of growing industries initiated the demand for engineers with mechatronical thinking, knowledge and actions; engineers who are comfortable in making the necessary decisions among a wide range of alternative approaches based on given design constraints. Such demand eventually guided universities toward creating dedicated degree programs for mechatronics. Such programs aim to develop an interdisciplinary way of thinking and practices to problem solving where the most effective engineering solution can be reached without bias from any given traditional engineering discipline. The question was how to develop an engineering curriculum and teach such a different philosophy within traditional engineering disciplines.

By the 1990s, mechatronics came to mean education and research identity as it emerged as an important engineering discipline. The most notable

features at this stage are the increased use of smart functions in mechatronical products and systems, miniaturization of the product, enhanced human–computer interaction and shortening the development cycle time by adopting the use of virtual prototyping and computer simulation. Closely related topics of development during the 1990s were: rapid prototyping; human–computer interaction; optoelectronics, electronic manufacturing and packaging; microelectromechanical systems; advanced manufacturing technology for polymer composite structures; knowledge-based systems; material handling technologies; etc. Furthermore, a new breed of intelligent components and systems start to emerge that combine an optimum blend of all available technologies featured by innovation, better quality, high reliability, better performance, compactness, and low cost. In addition, the consideration of the human factor during the design process led to ease of product use, safety and increased benefits for the end user. The advances of miniaturization technology led to the further development of microelectromechanical devices such as sensors and actuators and thereafter, the evolution of more sophisticated sensors and servomotors with built-in intelligence supported by local computer-based signal processing and control functions. Microprocessors embedded into mechanical systems led to size and efficiency gains with the birth of anti-lock brakes and electronic engine controls. Embedded systems and real-time software engineering was also accomplished through various types of embedded microcontrollers, which are an indispensable component of modern mechatronics systems.

The 1990s highlights the beginning of new era that merges, innovatively, mechatronics with modern communication and information technologies. These technologies were added to yield intelligent products that are portable, mobile and could be connected in large networks. This development made functions such as the remote operation of robots, home appliances, manufacturing, biomedical devices and health facilities possible. Furthermore, fundamental and applied developments in mini-, micro- and nanoscale electromechanics (especially microscale electromechanical systems (MEMS) and micro-opto-electromechanical systems (MOEMS)), control, informatics and power electronics have motivated and accelerated such growth and have found a wide range of applications. MEMS, such as the tiny silicon accelerometers that trigger automotive air bags, and other electronic engine controls are examples of the latter. After the mid-1990s, mechatronics has been widely recognized and has received wide attention due to its role and importance.

Since 2000, processor speeds, multi-core processors, advancements in building high memory capacity, smart sensors, the new era of micro- and nano-mechatronics predicated the boom in embedded and smart systems, such as in-car navigation systems, network-based audiovisual consumer electronic products, biomedical technologies as well as passive and active safety systems and more. The additional important trends are the increasing role of nanotechnology, and the incorporation of living and non-living molecules into processes and products, given their value in improving strength, appearance, durability, recycle and other functions.

Since the mid-2000s, HAFM represents the new era of mechatronics, which provides new methodologies and tools to design and to build new human-oriented machines and systems, featured by harmony and coexistence with its human user. HAFM allows creating, designing and support of new concepts for realizing intelligent human-oriented machines that coordinate and cooperate intelligently with their human users while understanding their status and surrounding situations through their interactive dynamic models supported by selected parameter constraints [HAB 12].

The interdisciplinary mechatronics engineering science field has experienced phenomenal growth since starting more than 42 years ago due to multidimensional advances in technology and methodology. Today, the term “mechatronics” encompasses a large array of technologies and it represents most of the research issues of modern design. The mechatronics design methodology is not only concerned with producing high-quality products, but maintaining these products as well. This area is referred to as lifecycle design which supports the knowledge-intensive engineering activities from initial conception to product disposal. The lifecycle design factors include delivery, reliability, maintainability, efficiency, serviceability, upgradeability, energy consumption and disposability (environmentally friendly) resulting in products that are designed from conception to retirement. Interdisciplinary mechatronics focuses on continuous individual self-learning, and team-based learning through a problem and project-based approach to fulfill the challenges of modern technology and the demand for innovation.

Modern engineering encompasses diverse interdisciplinary areas and interdisciplinary mechatronics engineering science has become and remains a significant design trend, which influences the nature of the product development process and technological changes, both in effect as well as pace. Yet, mechatronics was slow to gain industrial and academic acceptance, but this starts to change from early 2000, as witnessed by the growing

number of universities offering undergraduate and postgraduate mechatronics degree courses. Mechatronics has gained tremendous international attention and its teaching is becoming important, becoming widely and globally implemented. It has yet to be universally adopted as an engineering science discipline. Hence, it is important for both industry and academic institutions to work together in order to tune the required infrastructures that provide mechatronics engineers with essential interdisciplinary knowledge, practices and skills [HAB 06, HAB 07].

Interdisciplinary mechatronics engineering science focuses on continuous individual self-learning, and team-based learning through a problem and project-based approach to fulfill the challenges of modern technology and the demand for innovation. Figure 1.2 introduces the paradigm of the knowledge space describing a mechatronics engineering science discipline that reflects the authors understanding of the mechatronics knowledge space.

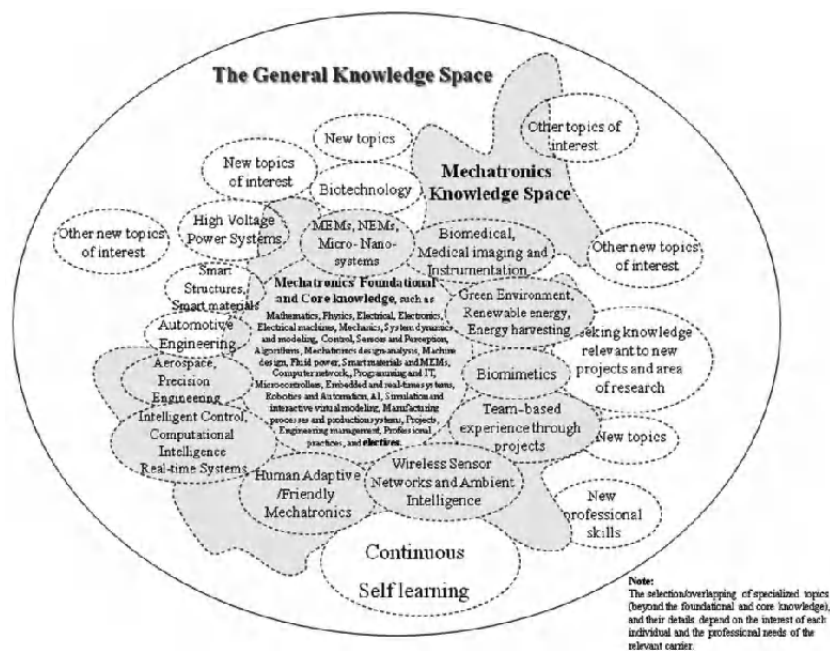


Figure 1.2. General and the mechatronics knowledge space paradigm.
Adopted and modified from [HAB 07]

1.2. Synergetic thinking, learning and innovation in mechatronics design

Mechatronics evolved into a philosophy supporting a new way of thinking and innovations in engineering science practices and design. It focuses on the learning process linked with actions rather than teaching. Thinking is a skill that can be developed and leads to effective learning. It is important to prepare students, engineers and members of research teams to engage in thinking and in knowing how to conduct an inquiry, how to listen and evaluate the opinions of others and how to compare alternatives. It is critical for successful thinking to associate two contradictory behaviors that should be separated in time: the aspiration and imagination to develop new ideas and how to argue these ideas to reach the best conclusion. This helps to organize thinking as a methodology and techniques to reach certain goals through a systematic use of thinking tools, clear awareness of the need for reflective thinking, self-evaluation of thinking, designing thinking tasks and methods to implement these tasks [HAB 07, HAB 08a, HAB 08b, HAB 12].

Interdisciplinary mechatronics engineering science is a challenging field that contributes to improve methodologies of the design process, aiming to achieve better efficiency, energy savings, ease of use and contributing human factors from the viewpoint of system integration. Mechatronics engineering is about mastering a multitude of disciplines, technologies and their interaction, whereas mechatronic science is about invention and development of new theories, models, concepts and tools in response to needs evolving from interacting scientific disciplines.

Imparting a synergetic and creative thinking through a problem-solving approach within the design process of mechatronical projects and systems requires not only changing the teaching methods and academic environment, but also adopting new assessment methods [MIO 98, WOL 89]. This component of the evaluation determines the degree to which the “across the curriculum, industrially relevant” approach to mechatronics is implemented, and maps each part of the design and implementation of a project activity on specified student learning outcomes.

These aspects of learning contribute to create challenges, curiosity, imagination and success in studying science and technology. It is evident that there is a role for science and technology education, as they remain a crucial part of general education [GAR 97]. Infusing creative thinking skills into the design process during project-based learning may provide a rich learning environment that contributes not only to the development of thinking skills

but also to a better understanding of the discipline under study. The mechatronics engineering science curriculum suggests that educators create rich learning environments filled with real-world applications. The assessment processes of learning outcomes in a rich learning environment has an important impact on the learning process.

It is important for the student to study and apply lateral thinking tools in order to deal with different alternatives, consider multiple factors, and to refrain from premature judgments on ideas. Lateral thinking refers to discovering new directions of thinking in the quest for a wealth of ideas through brainstorming. In addition, students should use vertical thinking tools in order to shortlist the best solutions, document their design process and then go through the details of implementation, testing and evaluation. Vertical thinking deals with the development of ideas' details and checks them against objective criteria. To be able to use both effectively, one must appreciate their differences. During the work on a technological project, lateral thinking initiates the learning process while students seek alternatives and examine different solutions through brainstorming [WAK 97, HAB 07]. Vertical thinking is essential at the stage of choosing a solution and developing it. Vertical thinking and lateral thinking complement each other, and both are the essential elements of creative thinking [DEB 86, HAB 08]. Lateral thinking is a central, but not singular, component of creative thinking.

Researchers, designers and commercial developers are now employing a wide range of advanced computing techniques that are capable of enhancing the working environment of designers and empowering engineers in their work [REI 99]. Some important elements of engineering informatics are explicit representations of physical and abstract components, symbolic and numerical process models, graphical user interfaces, large-scale databases and interactivity. The processing of information consists of low-level and high-level feedback control, supervision and diagnosis, and general process management. With the aid of the knowledge base and inference mechanisms, mechatronics systems with increasing intelligence are developed [ISE 97].

In mechatronics, software and sensory information represent vital dynamic elements that contribute to optimize capabilities, enhance smartness and improve flexibility of targeted systems through the design process. The capabilities of mechatronics systems are increasingly a function of their control software, and computational capabilities and limitations must be considered at every stage of design development and implementation. The software in mechatronics systems is made up of information, fused with methods and algorithms which make it possible to facilitate decision and

action support, and the building of associated intelligent functionalities. Information fusion can take place at different levels of abstraction in a system and at each subsystem as necessary. The resulting decision or action is in some sense better, qualitatively or quantitatively. In addition, utilizing the Internet and fusing mechatronics with information networks is supporting global connectivity and enabling people and systems to act and operate beyond the constraints of time and space. Just as the Internet induces us to think of information as a flow with a plurality of contexts, research on the network similarly forces us to think of science as a dynamic enterprise with a plurality of referents. Practicing interdisciplinarity means taking advantage of this plurality and guaranteeing the diffusion of ideas and the mobility of concepts [DAS 01, GAR 04, HAB 08a, HAB 12, HEW 96, REI 99].

For the purpose of avoiding unexpected trouble of a generalized design process, students/engineers/researchers should integrate the knowledge of the various disciplines, and learn the applicable standards and rules, as well as the underlying scientific principles and economic concepts with the necessary project management skill. Project-based learning in technology encourages students/engineers/researchers to work in teams where they can combine hands-on activities and project-based learning that could be used as a tool to develop students' competencies by working on integrated projects [BAR 02].

1.3. Human adaptive and friendly mechatronics

The evolution of mechatronics provides new methodologies and tools to design and build new machines and systems for safety, security and dependability. The main goals of modern technologies are to provide smart living and a clean environment that reduces operational stress, facilitates intelligent human life and to assist human well being while enhancing skills and maximizing performance. The increased functionality and smartness means the ability of products to identify and respond to changes in the human environment and to user's needs; that is to have intelligent machines that can be seen as a partner rather than just a tool. Hence, a key topic in the new mechatronics era is the design of human-oriented machines that implies intelligent and cooperative coexistence (beyond time and physical constraints) between technical and biological systems (human) within their natural environment.

A human and a machine may either share the same natural environment or be located at a different environment. If it's the latter case, it should be supported by mutual awareness to create a natural and mutual perception as

they are located within the same natural environment. Machines can overcome situations that could not be foreseen by the machine at the design stage in an autonomous way and up to a certain degree of complexity, while in critical situations it is necessary to facilitate, intelligently, the interaction with the human user/operator [HAB 12, SCH 96]. Mechatronics will meet science to enhance human knowledge, skills, intelligence and performance through the harmony of coexistence, and will help elderly/handicapped people to lead their lives safely and securely supported by modern science and technology. However, the design of human-oriented machines should bring no mental and psychological stress to its user. This will lead us to have a human-friendly and environmentally conscious mechatronic technology that coordinates work and cooperates with human beings. It appears natural to design machines that can cooperate in an intelligent way with their human users, thus enabling humans to make best use of the machines capabilities while it protects itself from any misuse or critical situations. Such human-oriented machinery will have novel features in their behavior relevant to the level of interaction with humans [HAB 06, HAB 07, HAB 12, HAR 05, SCH 96]. This implies that the enabling mechatronics systems have to understand human behavior and interact intelligently, adaptively, seamlessly and mutually for the sake to optimize the performance and both sides of the interaction should have the ability to assume the initiative within a task. Human-oriented machines should be able to adapt according to the competence of the human user and the capacity of adaptation may be related to the level of mental workload, and this implies that such machines have the ability to recognize a human's performance within the process. This can be achieved by using new methods and techniques with learning and adaptive properties that reflect the complexity of the working environment and the design requirements for the interactive cooperation between man and machine.

Robust HAFM systems adapt themselves to the skill level or dexterity of a human under various environments which aims to improve the user's skill, and assist the human-machine interaction to achieve the best performance. This includes not only assisting the control action, but also providing proper data and knowledge for understanding the situation and giving better decisions. Humans can improve the skill elements of their actions based on interactive signals that yield to change within skill parameters. Figure 1.3 shows the concept of HAFM [HAB 12].

To fulfill this reliably, it requires the following:

- 1) to model human and machine dynamics with special consideration on variable constraints;

- 2) to model the psychophysical characteristics of a human user/operator;
- 3) to model the required operation based on the demand for essential skills, knowledge and quality decisions;
- 4) The mechatronic system has to understand cognitive human behaviors, and extend assistance to the human by providing information that helps to understand the situation, which leads to the making of quality decisions and achieves a high level of coordination;
- 5) to enable dynamic interaction between such mechatronic systems and humans. Aspects that have to be considered in designing such interactive and cooperative machines are: work psychology, safety, ergonomics, control architecture, control modules, etc.;
- 6) that safety should be assured throughout the coupling and interactive relation between a human and machine with the capability to resolve urgent and critical situations.

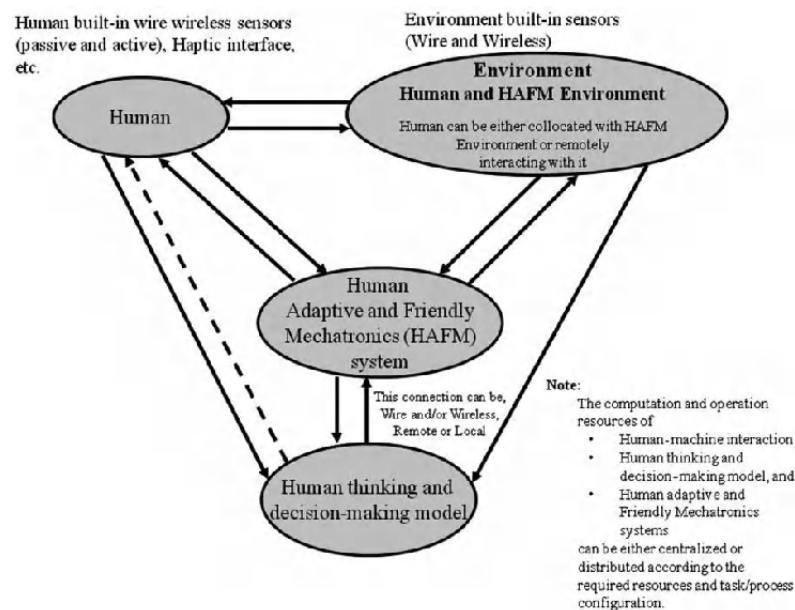


Figure 1.3. Conceptual diagram of human adaptive and friendly mechatronics.
Adopted and modified from [HAB 07]

There are number of criteria for how functions and authority are assigned to the human and machine depending on the versatility of the human and the consistency of the machine. These criteria may include techniques such as: leftover (with a high degree of automation the tasks, which cannot be automated are left over to the human), economic (for each partial task the cheapest solution is chosen), comparison (the task is allocated to the side who performs best) and complementary (assume that both sides have different capabilities, which complement each other) [HAB 07, SCH 96]. Understanding these criteria may help to develop intelligent tools that can assist human operators/users by giving them natural operational feelings. This needs to have physical symmetrical interaction between both sides mediated by common tools/modules, which includes their dynamical and behavioral/functional models, and is supported by sensory information reflecting each side's activities, and their local environments. Both sides may be located at the same local environment or located at completely different environments. Ambient intelligence, biologically inspired technology (Biomimetics), information and communication technologies, intelligent control and smart materials represent good candidates to facilitate the evolution of HAFM.

1.4. Conclusions

Mechatronics has opened up enormous technological possibilities that would never be possible to achieve or foresee by adopting a traditional single disciplinary or classical subsystem-based approach. Mechatronics has proven itself as a unifying interdisciplinary and intelligent engineering science paradigm as it is opening up new horizons and a bright future in all fields due to the stimulation of synergy, fusion and interdisciplinarity. This insight naturally lends itself to the concept of total quality that focuses on thinking, learning and innovation. HAFM represents the evolution of mechatronics into a new era that aims to provide new methodologies and tools to design and to build new human-oriented machines and systems characterized by harmony and coexistence with its human user. HAFM contributes to the change in the interactive relation between human and computerized systems, such as machines in order to find out the best solution to problems, maximize efficiency to achieve common goals and provide a good level of adaptation with some degree of autonomy for the machine side.

Mechatronics has gained tremendous international attention and its teaching is becoming important, and widely and globally implemented. The industry is now demanding engineers with a great, detailed and wide range of interdisciplinary knowledge, properly trained and supported by concurrent practices through a problem- and project-based curriculum to meet the challenges of interdisciplinary demand. To prepare capable mechatronics engineers, mechatronics curriculums and degree programs should feature a major design component through creative and innovative learning processes and balanced by basic and applied mechatronic theory. Accordingly, there is a need from the mechatronics community to focus their efforts on sustaining the progressive evolution of mechatronics as an engineering science discipline by strengthening its core pillars: interdisciplinarity, learning and thinking, knowledge fusion, problem- and project-based innovation, and skill building. In addition, mechatronics engineers and practitioners should always be associated with lifelong learning, and the capability to engage challenging problems and complex tasks with initiative.

1.5. Bibliography

- [BAR 02] BARLEX D., “The relationship between science and design and technology in the secondary school curriculum in England”, *Proceedings of the PATT12 Conference, March 15, 2002*, Eindhoven: University of Technology, pp. 3–12, 2002.
- [DAS 01] DASARATHY B.V., “Information fusion – what, where, why, when, and how?” *Information Fusion*, vol. 2, pp. 75–76, 2001.
- [DEB 86] DE BONO E., *The CoRT Thinking Program*, 2nd ed., Pergamon Press, Oxford, 1986.
- [GAR 04] GARBAY G., The role of information science in interdisciplinary research: a systemic approach, February 2004. Available at <http://www.interdisciplines.org/interdisciplinarity/papers/2>
- [GAR 97] GARDNER P., “The roots of technology and science: a philosophical and historical view”, *International Journal of Technology and Design Education*, vol. 7, pp. 13–20, 1997.
- [HAB 06] HABIB M.K., “Mechatronics engineering: the evolution, the needs and the challenges”, *Proceedings of the 32nd Annual Conference of the IEEE Industrial Electronics Society IEEE IECON 2006*, Paris, France, pp. 4510–4515, November 2006.

- [HAB 07] HABIB M.K., “Mechatronics: a unifying interdisciplinary and intelligent engineering paradigm”, *IEEE Industrial Electronics Magazine*, vol. 1, no. 2, pp. 12–24, 2007.
- [HAB 08a] HABIB M.K., “Interdisciplinary mechatronics: problem solving, creative thinking and concurrent design synergy”, *International Journal of Mechatronics and Manufacturing Systems*, vol. 1, no. 1, pp. 264–269, 2008.
- [HAB 08b] HABIB M.K., “Human adaptive and friendly mechatronics (HAFM)”, *IEEE International Conference on Mechatronics and Automation (ICMA 2008)*, 5–8 August, Takamatsu-Kagawa, Japan, pp. 61–65, 2008.
- [HAB 12] HABIB M.K., “Human adaptive and friendly mechatronics (HAFM), the evolution and the challenges”, *The 13th Mechatronics Forum International Conference*, Paper 5188, 17–19 September, Linz, Austria, 2012.
- [HAR 05] HARASHIMA F., “Human adaptive mechatronics”, *Proceedings of the 10th IEEE International Conference on Emerging Technologies and Factory Automation*, Italy, September 2005.
- [HEW 96] HEWIT J.R., King T.G., “Mechatronics design for product enhancement”, *IEEE/ASME Transactions on Mechatronics*, vol. 1, Issue 2, pp. 111–119, 1996.
- [ISE 97] ISERMANN R., “Mechatronics systems – a challenge for control engineering”, *Proceedings of the 1997 American Control Conference*, New Mexico, USA, pp. 2617–2632, 4–6 June 1997.
- [KYU 96] KYURA N., OHO H., “Mechatronics – an industrial perspective”, *IEEE/ASME Transactions on Mechatronics*, vol. 1, no. 1, pp. 10–15, 1996.
- [MIO 98] MIODUSER D., “Framework for the study of cognitive and curricular issues of technological problem solving”, *International Journal of Technology and Design Education*, vol. 8, pp. 167–184, 1998.
- [REI 99] REICH Y., SUBRAHMANIAN E., CUNNINGHAM D., *et al.*, “Building agility for developing agile design information systems”, *Research in Engineering Design*, vol. 11, pp. 67–83, 1999.
- [SCH 96] SCHWEITZER G., “Mechatronics for the design of human-oriented machines”, *IEEE-ASME Transactions on Mechatronics*, vol. 1, no. 2, pp. 120–126, 1996.
- [TRE 03] TREVELYAN J., “Introduction to mechatronics design”, 2003. Available at <http://www.mech.uwa.edu.au/mechatronics/index.htm>

- [WAK 97] WAKS S., “Lateral thinking and technology education”, *Journal of Science Education and Technology*, vol. 6, no. 4, pp. 245–255, 1997.
- [WOL 89] WOLF D., “Portfolio assessment: sampling pupil’s work”, *Educational Leadership*, vol. 45, no. 4, pp. 35–39, 1989.

Chapter 2

Micro-Nanomechatronics for Biological Cell Analysis and Assembly

2.1. Introduction of micro-nanomechatronics in biomedical fields

On the one hand technological advancement on the top-down fabrication process, or micro-machining, provides nanometer structures. On the other hand, the bottom-up fabrication process, or chemical synthesis such as self-assembly or supermolecule techniques, also provides nanometer structures. In fact, both approaches reach the nanometric scale with the present limitations of physical/chemical aspects. “Nanotechnology” has an important role in the combination of the top-down and bottom-up approaches. It is considered that the multi-scale technique: from the atomic scale to the meter scale will be realized in the near future.

The multi-scale technique is widely applied to the fields of research in the biological and medical fields (Figure 2.1). Against the background of an aging society, social demands to develop resource saving technology and highly efficient energy systems required the integration of the academic disciplines of engineering, biology and the medical sector with multi-scale technology. This integration will be applied for various applications. In the engineering field, the bio-micro-nanomimetics system and biologically driven micro-nanosystem are investigated to realize the system in semi-set or

Chapter written by Toshio FUKUDA, Masahiro NAKAJIMA, Masaru TAKEUCHI, Tao YUE and Hirotaka TAJIMA.

wet environments. The biological system has quite a highly efficient energy consumption system, for example a flagellar motor of bacteria has almost 100% energy efficiency [MOR 09]. The current mechatronics system will be inactive enough to take into account such a biological system. In the biological field, the single cell analysis system and single cell nanosurgery system are investigated as an evolutionary technology to analyze/manipulate a single cell. Current group analysis can be obtained for the averaged information as a cell group. The micro-nano robotics control technique will be achieved to obtain individual information or manipulation in single cells. In the medical field, the regeneration medicine, tissue medicine and alternative healthcare are readily investigated.

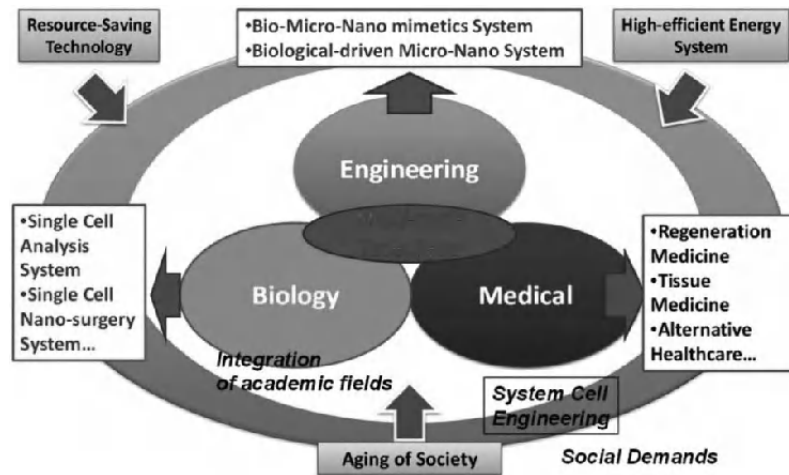


Figure 2.1. Micro-nano robotic control techniques for biological/medical applications

The current direction of research is moving to synthesis from analysis of the biological system as shown in Figure 2.2. The multi-scale technique is developed for cell analysis in single cell/group cell/tissue levels. The significant improvement of regeneration engineering/biology will be achieved by three-dimensional construction *in vitro* from “*in vivo*” biological functions. Stem cell engineering is a key technology to realize tissue/organs developed by embryo stem cells (ES cells). By reducing the snit factors of the induced pluripotent stem cells (iPS cells), to be changed to cancer cells, this research field will be dramatically improved in the near future [YEE 06].

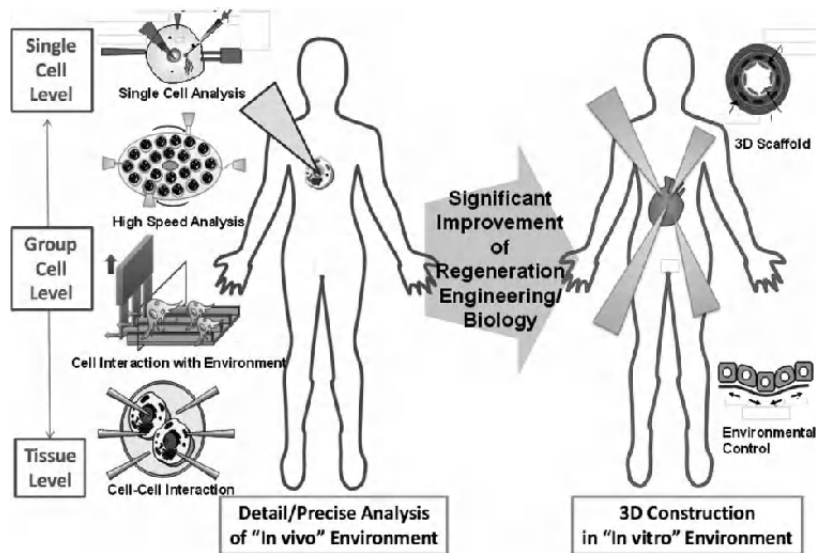


Figure 2.2. *The schematics of construction of in vitro system from analysis*

2.2. Configuration of micro-nanomechatronics

Mechatronics is an interdisciplinary field through synergetic integration of mechanical and electrical engineering with computer technology. For example, in the field of robotics, industrial robots are actuated by replacing the hydraulic actuator with electric motors. Miniaturizing technologies, such as micro-nanotechnology, grows dramatically in the engineering field around 2000. These technologies are commonly used in our daily lives, in applications for automobiles, computer peripherals, printers, cameras, amusements, robotics, automation, environmental monitoring, energy resource, biological/medical treatments and so on. Currently, the micro-nanomechatronics field is set out to improve device specification to have more greater efficiency, higher integration, higher functionality, lower energy consumption, lower cost, miniaturization and so on.

The relationship of required technologies for micro-nanomechatronics is shown in Figure 2.3. The material/sensing/actuation technologies are the fundamental level for micro-nanomechatronics. Their scientific discoveries

or technological developments are important to bring innovations in this field. Control technology is constructed from actuation and sensing technologies, but we need to consider the energy supply system. The fabrication technology is strongly related with material technology. Finally, the assembly technology is achieved on the basis of these technologies and makes a feedback loop from the micro-nano evaluation technology to check the requirements, safety issues, improvement of the system and so on. Through this flows of research configuration techniques are constructed for micro-nanomechatronics.

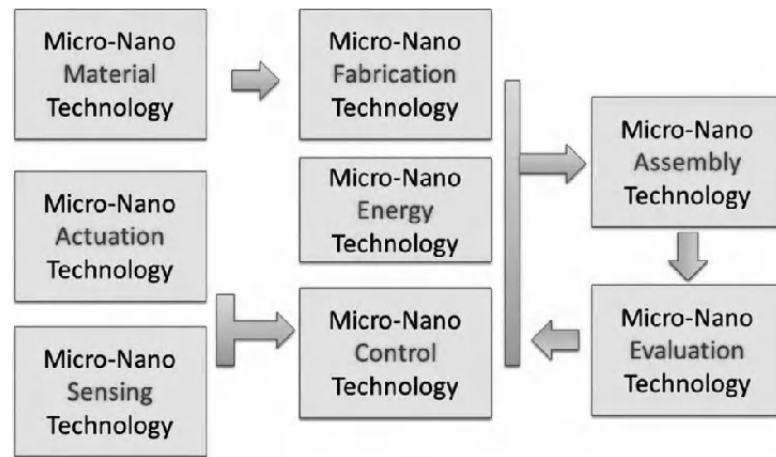


Figure 2.3. *Technological relationship for micro-nanomechatronics system*

The examples for each technique are summarized in Figure 2.4. Various methods are developed and investigated for each technique. To establish micro-nanomechatronics, these need to be selected and integrated on a case-by-case basis for the applications. From the application viewpoint, the environmental constraint is one of the issues when choosing the techniques. For example, the conditions of temperature, gas pressure, vibrations and wet/dry are important and need to be considered. For the biological application, the environmental condition should be tougher including the sterile/biocompatible/biodegradable conditions. Normally, biological cells are quite sensitive to the environment, and it is required to design the system with the environmental constraints factored in for biomedical applications.

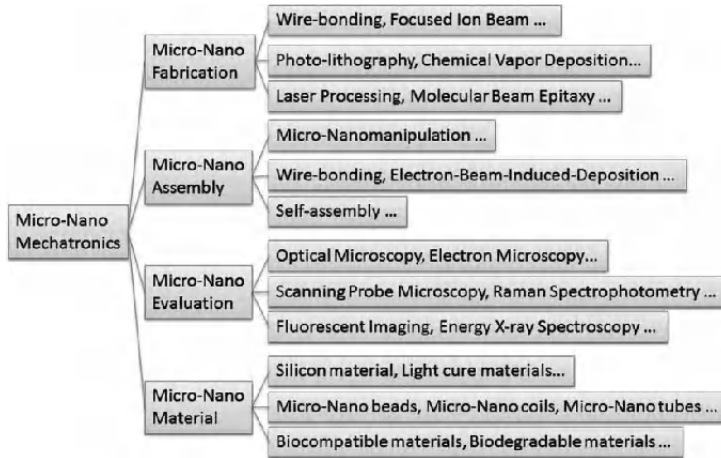


Figure 2.4. Examples of techniques on micro-nanomechanics

Examples of actuation mechanisms are summarized in Table 2.1. From the viewpoint of physical interactions, the actuation technology is mainly divided into contact or non-contact ways. For a micro-nanomanipulation system, the piezo-driven mechanisms are widely used because of their good performance in the positioning accuracy. For biological cell manipulation, the manipulation system is required to approach the targeted cells under fluidics, which is needed to keep the cells in a viable condition. Through contact manipulation, the fluidics surrounding the targeted cell is affected during manipulation, making some turbulence. On the other hand, the non-contact manipulation mechanism is effectively used for biological cell manipulation in wet environments without direct physical contact to the fluidics. The electrical and magnetically driven mechanisms have larger actuation forces by comparison with optical manipulation. However, optical manipulation is superior on the local actuation of a targeted object by its narrower actuation range.

Examples of sensing mechanisms are summarized in Table 2.2. The local and highly accurate sensing mechanisms are important for the micro-nanomechanics applications. To select the sensing mechanism, the various limitations need to be considered; such as (1) environments (wet/dry/vacuum, etc., sterile or non-sterile environments, etc), (2) space and (3) energy supply limitations. To construct the control system, the sensing mechanism needs to be combinable with the actuation mechanism as an micro-nanomechanics system.

Micro-nano actuation mechanism	Physical interaction	Accuracy	Actuation range	Durability	Environmental condition
Electrical motor driven	Contact	++	+++	+++	Dry
Hydraulic pressure driven	Contact	++	+++	+	Dry
Piezo-driven	Contact	+++	+	++	Dry
Biological cell driven	Contact	+	+	+	Wet
Bacteria driven	Contact	+	+	+	Wet
Optically driven	Non-contact	+++	+	++	Dry, wet
Electric driven	Non-contact	++	++	++	Dry, wet
Magnetically driven	Non-contact	++	++	++	Dry, wet

Table 2.1. *Examples of actuation mechanisms for micro-nanomechatronics*

Micro-nano sensing categories	Examples
Positioning sensing	Laser meter, ultrasonic sound meter, piezo sensing, microscopic imaging, etc.
Force sensing	Strain gauge, electric capacitance, electrostatic gauge, piezo cantilever, etc.
Electromagnetic sensing	Flowmeter, particles charges, photodetectors, etc.
Chemical sensing	Gas sensing, ionic sensing, biosensing, bioindicator, etc.
pH Sensing	Indicator, potential difference mechanism, field effect transistor, etc.
Temperature sensing	Indicator, infrared intensity, resistive change, etc.
Cell activity sensing	Indicator, fluorescent dye, fluorescent protein, physical/chemical interaction, etc.

Table 2.2. *Examples of sensing mechanisms for micro-nanomechatronics*

2.3. Micro-nanomechatronics for single cell analysis

The micromanipulation of biomicroparticles such as microorganisms and cells has become essential in various industries, such as in the food industry, the medical industry, for environmental purification and so on. Many microorganisms such as oil-degrading bacterium, basophilic bacterium and thermophile bacterium have already been discovered. By using the useful characteristics of these microorganisms, the development of new medicine, the production of useful materials and environmental purification has been achieved. For example, the recombination of microorganisms are used for the production of biological plastic (polyhydroxyalkanoate (PHA)) [ABE 10] or the degradation of biodegradable plastics (poly-butylene succinate-co-adipate (PBSA)) [KIT 11]. These are examples of microorganism application to the post-petrochemical industry amid energy problems such as a decrease in oil resources.

However, microorganisms which have so far been discovered are estimated as less than 10% of the total microorganisms. For the remarkable advancement of microbiology and bioengineering, it is important to develop a new technology of manipulation and separation and discover unknown microorganisms and their characteristics. Much research on cell analysis has been done for the investigation of the unknown properties of them. There are several different methods to understand the biological properties of cells. For example, the group cell analysis and single cell analysis are based on the direct observation of each specimen. Both methods are important, but each method has its own characteristics. With the former method, the time-course of cell properties such as an increase in cell volume and DNA synthesis during the division cycle has been studied. Group cell analysis is very important for application in industry. However, it is only possible to obtain general information on each cell.

In recent years, single cell analysis has received much attention with the development of biotechnology [WAK 05, ROM 07]. For the actual microbial community, it is a heterogeneous population as shown in Figure 2.5. After stable phase through the exponential growth phase, this population divides into three phases such as viable but not culturable, apoptosis and adaptation to a new environment. Essentially, the cell population is not homogeneous, and the mechanism of differentiation to the heterogeneous is not known. Microbial activity is not uniform, and the mechanism of gene expression is not known. In addition, gene expression is dynamically changing, and its

property is unknown. For single cell analysis, an extraction of a single target cell from a large number of samples and continuous monitoring of the cell are necessary. For these reasons, “biomanipulation” is quite an important technique, which is defined as the technique for manipulation of biological cells from the micro-nano scale to the macro scale, based on single cell manipulation and the observation of single cell reactions.

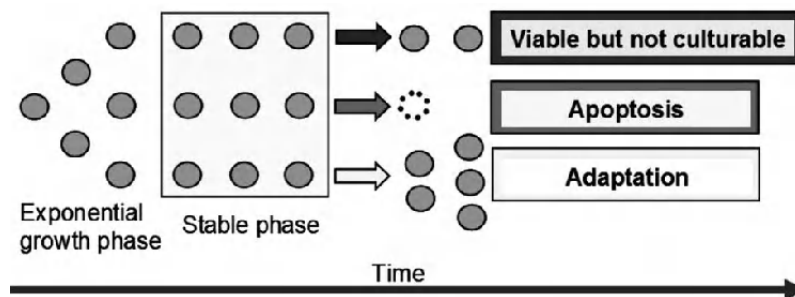


Figure 2.5. *Mechanism of microbial community*

The major examples of biomanipulation techniques are categorized in Table 2.3. The purposes of biomanipulation are classified into major three directions. The first direction is the “analysis of biological cells” such as “(1) mechanical interaction, (2) cell–cell interaction and (3) measurement/imaging”. The second direction is “manipulation of biological cells” such as in “(4) gene manipulation/injection, (5) separations/immobilization, (6) environment control and (7) molecular manipulation”. The third direction is “assembly of biological cells” such as “(8) shape control of cell/tissue and (9) 2D/3D cell assembly”. Until now, the contribution of “biomanipulation” has mostly been the investigation of analysis and manipulation applications. Recently, based on this knowledge and techniques, the assembly applications are given greater attention as a current research topic.

In the following sections in this chapter, we introduce our recent results on the semi-closed microchip for the single cell analysis and biological cell assembly using photo-linkable resin based on single cell analysis techniques.

Purpose	Techniques	Examples	Issues
1) Mechanical interaction	Biomechanics, differentiation, growth condition, etc.	Mico-nano handling, μ TAS, BioMEMS/NEMS, etc.	Local control, 3D positioning, 3D force sensing, individual/local sensing, etc.
2) Cell-cell interaction	Intercellular materials, protein-protein interaction, etc.	Micro-nano probe, regeneration engineering, etc.	Local sensing, Minimally invasive technique, <i>in vitro</i> realization of <i>in vivo</i> system, etc.
3) Measurement/imaging	Real-time measurement, gene expression, etc.	AFM, nano-probe optical tweezers, etc.	Real-time imaging, local force/torque/positioning measurement, etc.
4) Gene manipulation/injection	Stability in the cell transportation, expression, etc.	Micro-nano-probe, injection, μ TAS, BioMEMS/NEMS, etc.	Viability, high speed, mass production, high efficiency, gene expression, etc.
5) Separations/Immobilization	On-chip cell analysis, gene expression, etc.	Optical tweezers, DEP, electrophoresis, etc.	High speed, <i>in situ</i> , low damage, etc.
6) Environmental control	Chemical, mechanical, electrical stimulation, mechanical stimulation, etc.	Micro-nano-probe, μ TAS, BioMEMS/NEMS, etc.	Local control, density gradient real time, probes, etc.
7) Molecular manipulation	Cell modeling, functional protein synthesis, etc.	Optical tweezers, nano-probe, electrophoresis, magnetically driven, etc.	Protein synthesis, mutation, functional control, etc.
8) Shape control of cell/tissue	Chemical control of cell surface, shape and pattern control, etc.	Micro-nano fabrication, SPM, μ TAS, BioMEMS/NEMS, etc.	Local chemical control, cell selection/shape/pattern, function control, etc.
9) 2D/3D Cell assembly	Control of extracellular matrix/cytokine, etc.	Scaffold, bioprinting, spheroid, cell origami, cell sheet engineering, etc.	High speed, high efficiency, complex tissue organization, etc.

Table 2.3. Examples of biomanipulation techniques

2.4. Semi-closed microchip for single cell analysis

For single cell analysis, two types of systems have mainly been investigated. One is the open chambers such as a petri dish [TAK 08, INO 08] and the other is the closed chambers such as a microchip [HUN 05, WAN 10]. The open chambers allows probe manipulation. Probe manipulation is one of the most important techniques for single cell analysis, because probe-type devices realize manipulating arbitrary single cell and controlling the environment of the arbitrary position. The probe-type devices also perform the measurement of cellular status [CLA 07], for example temperature distribution. However, it is difficult to prevent the effects of the external environment, such as evaporation of a solution and contamination when the open chambers are used.

On the other hand, the closed chambers can prevent the effects caused by the external environment. However, it is difficult to use the probe-type devices on conventional microchips. Generally, non-contact manipulation techniques like optical tweezers [MAR 03, ARA 04] or fluid force [UVE 09, SUN 06] are needed for cell manipulation on the microchip. These non-contact manipulation techniques have some disadvantages. For example, the optical tweezers can only apply a pN order force to cells, and it is difficult to manipulate large cells like an egg cell. It is also difficult to manipulate the target single cell when the fluid force is used to manipulate cells. The closed chambers have to fabricate complex microchannels [CAR 06, TAN 07] before experiments to use them for single cell analysis.

Hence, the desire to develop new devices that have the advantages of both open and closed chambers. We proposed a semi-closed microchip, which realizes the probe manipulation and keeping the seal [TAK 10]. Figure 2.6 shows the schematic of a microchip for cell manipulation. Cell culture, cell analysis and target cell harvesting can be realized continuously in the semi-closed microchip under various environments. The proposed semi-closed microchip is shown in Figure 2.7. The microchannel is fabricated in the microchip to realize the exchange of solution in the bath. To realize the probe manipulation, the semi-closed microchip has the bath in the middle of microchannel. The probe-type devices can be inserted into the microchip through this bath. The bath is sealed off by thin oil film, which can prevent the evaporation of the solution in the bath. The probe-type devices can be inserted and taken out, still keeping the seal of the bath.

To extend the semi-closed microchip as a useful single cell analysis system, it is necessary to develop and integrate other devices such as a pump,

a probe-type device and filter. We developed the probe-type devices. The sizes of cells are generally 10–100 μm , and the volumes are less than 1 nL. So it is necessary to develop the probe-type device that can control the spout and suction volume easily in nL in order to realize single cell manipulation under a microscope. We proposed a novel probe-type device “thermal gel-actuated probe”, which uses thermal gel as an actuator to realize the spout and suction in an nL order. The spout and suction can only be controlled to switch the heater on and off. The thermal gel-actuated probe can be fabricated at a low cost; so disposable use can be realized by the thermal gel-actuated probe.

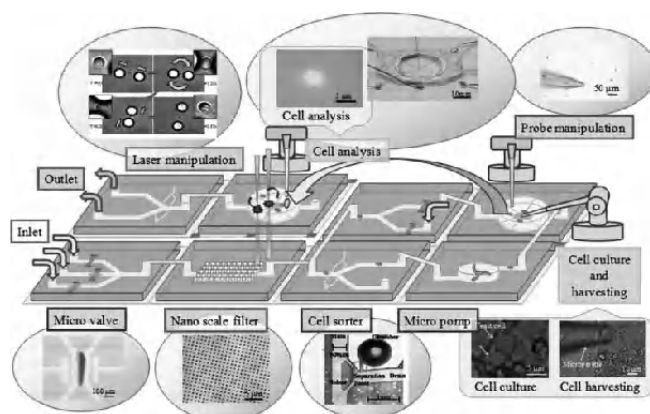


Figure 2.6. Configuration of microchip for cell manipulations. For a color version of this figure please see www.iste.co.uk/mechatroneng.zip

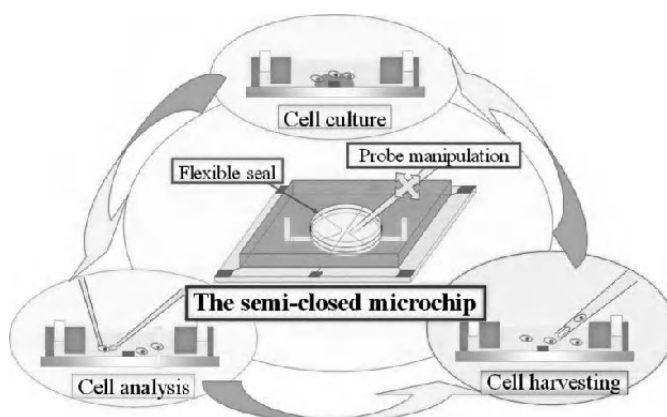


Figure 2.7. Conceptual figure of semi-closed microchip

Cell harvesting by the thermal gel-actuated probe was demonstrated under the microscope using a semi-closed microchip. Indium tin oxide (ITO) electrodes are fabricated as heaters at the center of the bath. A poly-(N-isopropylacrylamide) (PNIPAAm) solution, which shows a gelation more than 32°C , can be gelled by applying voltages to these electrodes. Cells in the semi-closed microchip can be fixed by this generated thermal gel. This cell fixation realizes the stable observation under a microscope during cell culture and cell analysis in the semi-closed microchip. The thermal gel-actuated probe was set on the micromanipulator. The suction and the spout of the solution were controlled only to switch the heater on and off. Figure 2.8 shows the experimental results. The thermal gel-actuated probe was attached to the target yeast cells at first. After positioning of the thermal gel-actuated probe, the target yeast cells were harvested. This result indicates that the thermal gel-actuated probe can harvest the target cells that are fixed by the thermal gel.

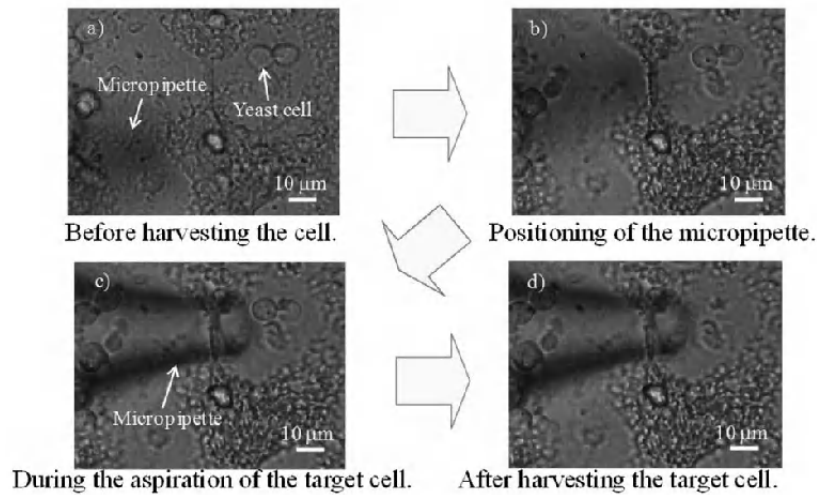


Figure 2.8. Single cell harvesting using semi-closed microchip

2.5. Biological cell assembly using photo-linkable resin based on the single cell analysis techniques

The cells inside real tissues and organs are usually arranged with a large amount according to certain patterns and shapes, such as neural cells, skin and blood vessels [TAK 10]. In tissue engineering, important issues are cell

encapsulation inside certain structures and high-throughput assembly of these structures [USH 02].

There are many kinds of cell manipulation methods, such as surface adhesion, optical tweezers and dielectrophoresis (DEP). By using surface adhesion, it is possible to manipulate cells without damaging the cells. The position of cells is not flexible because surface adhesion depends on the surface property, which is not easy to change [SUU 05]. Optical tweezers is a low-damage approach and easily controlled. An automatic cell manipulation system is constructed. But the manipulation force is weak as compared with the flow resistance [LAN 93]. Compared with other methods of cell manipulation, DEP is easier to control, and it is non-contact, which means less damage to the cells. Because of the low physiological stress caused by DEP, the cells remain viable after treatment and can be cultured for further purposes [BER 04]. By DEP force, it is possible to manipulate particles with great selectivity. It is possible to allow the separation of cells or the orientation and manipulation of nanoparticles and nanowires. Consequently, DEP force is widely used in cell manipulation, such as sorting, separation and patterning [HU 11].

The convenient methods for cell immobilization are based on such an aspiration, pressure of solution and fluidic structure [JAE 08]. The advantage of aspiration and pressure is that the fixing force is large, whereas the disadvantage is in damaging the cells [VAL 10]. By special fluidic structures, cells are immobilized inside a microfluidic chip. The immobilized cells are difficult to be analyzed further [TSU 10]. On-chip fabrication based on photo-crosslinkable resin via UV illumination is a creative way for immobilizing cells. Cells are directly immobilized inside the microfluidic chip. There are several advantages, such as high speed, low cost and their arbitrary shape [TIX 04].

To culture the cells in a tissue, taking account of the deficiency of oxygen and nutrients and the accumulation of wastes, thus the survival rates of cells are quite important issue. Basically, these problems are not easy to solve *in vitro*. The conservation limit of the thickness of stacks of a cell sheet by diffusion of oxygen, nutrients and wastes is less than 200 μm [DI 06]. Hence, a blood vessel structure is important to construct with three-dimensional tissue.

For tissue engineering, it is required to construct cell patterns and immobilize patterned cells inside certain structures. We proposed a novel method of forming a cell pattern through the DEP forces and immobilizing

cells by photo-crosslinkable resin inside microfluidic chips as shown in Figure 2.9 [YUE 11]. High-speed cell manipulation, including patterning and concentration control by DEP was demonstrated. Movable microstructure embedding cells is on-chip fabricated.

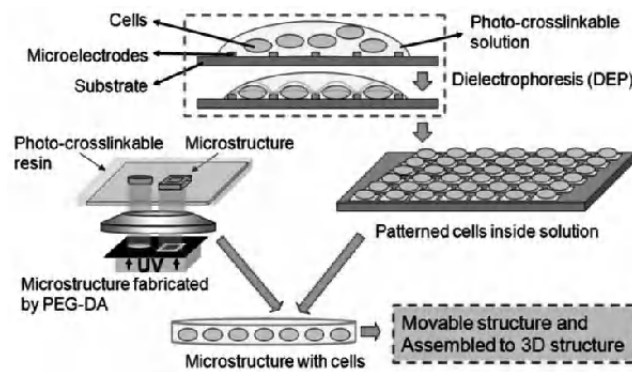


Figure 2.9. A schematic drawing of the cell assembly method

The experimental results show that the cell line patterns that contain hundreds of yeast cells can be formed by DEP within 1 s as shown in Figure 2.10. By applying DEP force, several microelectrodes are fabricated by ITO and Cr/Au, which are coated on the glass. The two kinds of DEP responses for yeast cell (W303) and other particles were confirmed experimentally. Based on the negative DEP phenomenon, cell traps generated by microelectrode are demonstrated. Position control and transportation of yeast cells are performed by using cell traps.

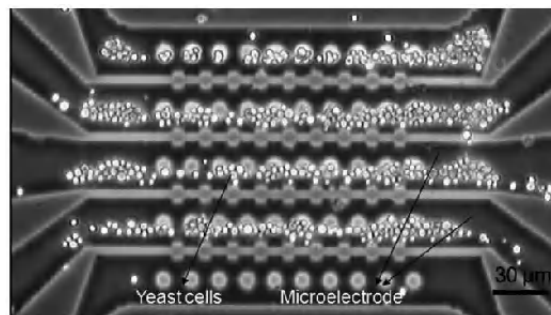


Figure 2.10. The line pattern of yeast cells formed by n-DEP based on ITO microelectrode

Figure 2.11 shows the assembly of a microstructure embedded with line-patterned cells. Figures 2.11(a) and 2.11(b) show the micro-channel before cleaning away the cells outside the microstructure. Figures 2.11(c) and 2.11(d) show the microstructures that contain three lines of yeast cell inside micro-channel after cleaning. The on-chip fabrication for arbitrary shapes of microstructures was performed using polyethylene glycol diacrylate (PEG-DA), which is the photo-linkable resin. With the cell patterning by DEP and immobilizing by on-chip fabrication, a microstructure that contains three lines of yeast cells is fabricated in the microfluidic channel, inside a PEG-DA and NaCl solution. It is designed for movable microstructures embedding microbeads of which, the concentration is controllable.

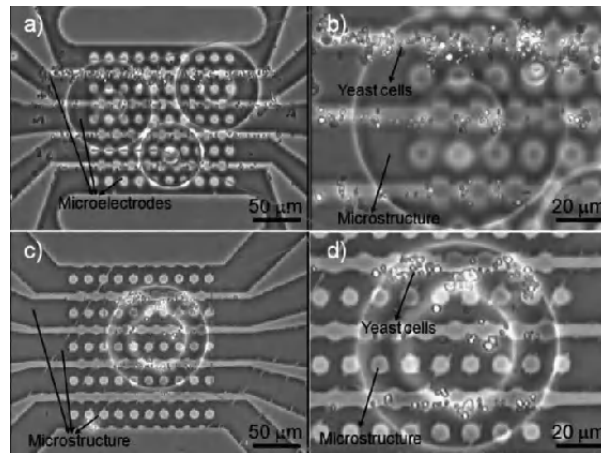


Figure 2.11. Micro-structure embedded with line-patterned cells. For a color version of this figure please see www.iste.co.uk/mechatroneng.zip

2.6. Conclusion

In this chapter, we introduced micro-nanomechatronics for biological cell analysis and assembly. The micro-nanomanipulation technique is a key technique to analyze and synthesize a biological system *in vitro*. We introduced the semi-closed microchip for the single cell analysis with a thermal gel-actuated probe. The biological cell assembly was also introduced by the DEP forces and immobilizing cells by photo-crosslinkable resin inside microfluidic chips. These research fields will be continuously investigated, becoming more sophisticated for realizing the various biomedical applications *in vitro* from the “*in vivo*” biological functions.

2.7. Acknowledgments

We would like to thank Professors Michio Homma, Seiji Kojima and Masaru Kojima for various discussions on biological aspects and Toshifumi Inada for providing us with the yeast wild-type strain W303 cells for the experiments conducted. This work was partially supported by the global COE program “COE for Education and Research of Micro-Nanomechatronics” of Nagoya University and a grant-in-aid for scientific research from the Ministry of Education, Culture, Sports, Science and Technology of Japan.

2.8. Bibliography

- [ABE 10] ABED M., SUDESH K., “Bacterially produced polyhydroxyalkanoate (PHA): converting, renewable resources into bioplastics current research”, *Technology and Education Topics in Applied Microbiology and Microbial Biotechnology*, vol. 2, pp. 1395–1404, 2010.
- [ARA 04] ARAI F., YOSHIKAWA K., SAKAMI T., *et al.*, “Synchronized laser micromanipulation of multiple targets along each trajectory by single laser,” *Applied Physics Letters*, vol. 85, pp. 4301–4303, 2004.
- [BER 04] BERNSTEIN, R.W., SCOTT M., SOLGAARD O., “BioMEMS for high-throughput handling and microinjection of embryos”, *Mems/Moems Technologies and Applications II*, vol. 5641, pp. 67–73, 2004.
- [CAR 06] CARLO D.D., WU L.Y., LEE L.P., “Dynamic single cell culture array,” *Lab on a Chip*, vol. 6, pp. 1445–1449, 2006.
- [CLA 07] CLARKE R.W., PIPER J.D., YING L., *et al.*, “Surface conductivity of biological macromolecules measured by nanopipette dielectrophoresis,” *Physical Review Letters*, vol. 98, p. 198102, 2007.
- [DI 06] DI CARLO D., AGHDAM N., LEE L.P., “Single-cell enzyme concentrations, kinetics, and inhibition analysis using high-density hydrodynamic cell isolation arrays”, *Analytical Chemistry*, vol. 78, no. 14, pp. 4925–4930, 2006.
- [HU 11] HU S., SUN D., “Automatic transportation of biological cells with a robot-tweezer manipulation system”, *The International Journal of Robotic Research*, vol. 30, no. 14, pp. 1681–1694, 2011.
- [HUN 05] HUNG P.J., LEE P.J., SABOUNCHI P., *et al.*, “Continuous perfusion microfluidic cell culture array for high-throughput cell-based assays,” *Biotechnology and Bioengineering*, vol. 89, no. 1, pp.1–8, 2005.
- [INO 08] INOUE K., TANIKAWA T., ARAI T., “Micro-manipulation system with a two-fingered micro-hand and its potential application in bioscience”, *Journal of Biotechnology*, vol. 133, pp. 219–224, 2008.

- [JAE 08] JAEGER M.S., UHLIG K., SCHNELLE T., *et al.*, “Contact-free single-cell cultivation by negative dielectrophoresis”, *Journal of Physics D-Applied Physics*, vol. 41, p. 175502, 2008.
- [KIT 11] KITAMOTO H.K., SHINOZAKI Y., CAO X.H., *et al.*, “Phyllosphere yeasts rapidly break down biodegradable plastics”, *AMB Express* vol. 1, no. 44, pp. 11, 2011.
- [LAN 93] LANGER R., VACANTI J.P., “Tissue engineering”, *Science*, vol. 260, pp. 920–926, 1993.
- [MAR 03] MARUO S., IKUTA K., KOROGI H., “Submicron manipulation tools driven by light in a liquid”, *Applied Physics Letters*, vol. 82, pp.133–135, 2003.
- [MOR 09] MORA T., YU H., SOWA Y., *et al.*, “Steps in the bacterial flagellar motor”, *PLoS Computational Biology*, vol. 5, no. 10, p. e1000540, 2009.
- [ROM 07] ROMAN G.T., CHEN Y., VIBERG P., *et al.*, “Single-cell manipulation and analysis using microfluidic devices”, *Analytical and Bioanalytical Chemistry*, vol. 387, pp. 9–12, 2007.
- [SUN 06] SUN Y., YIN X.F., “Novel multi-depth microfluidic chip for single cell analysis”, *Journal of Chromatography A*, vol. 1117, pp. 228–233, 2006.
- [SUU 05] SUURONEN E.J., SHEARDOWN H., NEWMAN K.D., *et al.*, “Building in vitro models of organs”, *International Review of Cytology*, vol. 244, pp. 137–173, 2005.
- [TAK 08] TAKAMATSU H., UCHIDA S., MATSUDA T., “In situ harvesting of adhered target cells using thermoresponsive substrate under a microscope: principle and instrumentation”, *Journal of Biotechnology*, vol. 134, pp. 297–304, 2008.
- [TAK 10] TAKEUCHI M., NAKAJIMA M., KOJIMA M., *et al.*, “Nanoliters discharge/suction by thermoresponsive polymer actuated probe and applied for single cell manipulation”, *Journal of Robotics and Mechatronics*, vol. 22, no. 5, pp. 644–650, 2010.
- [TAN 07] TAN W.H., TAKEUCHI S., “A trap-and-release integrated microfluidic system for dynamic microarray applications”, *Proceedings of the National Academy of Sciences*, vol. 104, pp. 1146–1151, 2007.
- [TIX 04] TIXIER-MITA A., CHIRAL M., OSTROVIDOV S., *et al.*, “A silicon micro-system for parallel gene transfection into arrayed cells”, *Proceedings of the uTAS 2004 Symposium*, The Royal Society of Chemistry, pp. 180–182, 2004.
- [TSU 10] TSUTSUI H., YU E., MARQUINA S., *et al.*, “Efficient dielectrophoretic patterning of embryonic stem cells in energy landscapes defined by hydrogel geometries”, *Annals of Biomedical Engineering*, vol. 38, pp. 3777–3788, 2010.

- [USH 02] USHJIMA H., ISHIDA K., NAGASHIMA H., “Bovine nucleus transplantation by intracytoplasmic injection”, *Journal of Reproduction and Development*, vol. 48, pp. 619–626, 2002.
- [UVE 09] UVET H., HASEGAWA A., OHARA K., *et al.*, “Vision-based automated single-cell loading and supply system”, *IEEE Transactions on Nanobioscience*, vol. 8, pp. 332–340, 2009.
- [VAL 10] VALERO A., BRASCHLER T., DEMIERRE N., *et al.*, “A miniaturized continuous dielectrophoretic cell sorter and its applications”, *Biomicrofluidics*, vol. 4, no. 2, 2010.
- [WAK 05] WAKAMOTO Y., RAMSDEN J., YASUDA K., “Single-cell growth and division dynamics showing epigenetic correlations”, *Analyst*, vol. 130, pp. 311–317, 2005.
- [WAN 10] WANG Y., CHEN Z.Z., LI Q.L., “Microfluidic techniques for dynamic single-cell analysis”, *Mikrochimica Acta*, vol. 168, pp. 177–195, 2010.
- [YEE 06] YEE CHEE J., YOGA S.S., LAU N.S., *et al.*, “Induction of pluripotent stem cells from mouse embryonic and adult fibroblast cultures by defined factors”, *Cell*, vol. 126, no. 4, pp. 663–676, 2006.
- [YUE 11] YUE T., NAKAJIMA M., ITO M., *et al.*, “High speed laser manipulation of on-chip fabricated microstructures by replacing solution inside microfluidic channel”, *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2011)*, pp. 433–438, 2011.

Chapter 3

Biologically Inspired CPG-Based Locomotion Control System of a Biped Robot Using Nonlinear Oscillators with Phase Resetting

This chapter deals with a design for a locomotion control system of a biped robot inspired by the physiological concept of a central pattern generator (CPG). It shows the usefulness of the control system by stability analysis using a simple biped robot model and by various experiments using actual biped robots.

3.1. Introduction

In the robotics field, interest in the study of robots with legs has increased. However, these robots still have difficulties in establishing adaptive locomotor behaviors in various situations. In contrast, humans and animals produce adaptive walking in diverse environments by cooperatively and skillfully manipulating their complicated and redundant musculoskeletal systems, where neuromechanical interaction is crucial. To create new control strategies for legged robots, it is natural to use ideas inspired from biological systems. For that purpose, elucidating the mechanisms for creating adaptive locomotor

behaviors in biological systems and constructing design principles to produce the adaptability in robotic systems are crucial issues.

Physiological studies have shown that the CPG in the spinal cord greatly contributes to rhythmic limb movement, such as locomotion [GRI 75, ORL 99, SHI 76]. So far, based on the physiological concept of the CPG, locomotion control systems of legged robots have been developed to create adaptive walking of the robots in various environments [IJS 07, IJS 08, LIU 08, KIM 07, NAK 04, NAK 06, NOM 09, STE 10].

The CPG can produce oscillatory behaviors even without rhythmic input and proprioceptive feedback. However, it must use sensory feedback to produce effective locomotor behavior. For example, spinal cats produce locomotor behaviors on a treadmill and their gait changes depending on the belt speed [FOR 73, ORL 99]. This result suggests that the tactile sensory information between their feet and belt influences the locomotion phase and its rhythm generated by the CPG [DUY 00]. Physiological evidence has shown that the locomotion rhythm and its phase are modulated by producing phase shift and rhythm resetting based on sensory afferents and perturbations (phase resetting) [CON 87, DUY 77, GUE 95, LAF 05, SCH 98]. Moreover, the functional roles of phase resetting in the generation of adaptive walking have been investigated using neuromusculoskeletal models of biological systems [AOI 10, YAK 04, YAM 03a, YAM 03b].

In this study, we design a simple locomotion control system for a biped robot using nonlinear oscillators based on the physiological concept of the CPG and physiological evidence of phase resetting. This control system produces adaptive locomotion of a biped robot. This study shows the usefulness of this locomotion control system from stability analysis using a simple biped robot model and from various experiments using actual biped robots.

3.2. Locomotion control system using nonlinear oscillators

3.2.1. CPG-based locomotion control system

The organization of the CPG in biological systems remains largely unclear. However, recent physiological findings suggest that the CPG consists of hierarchical networks composed of a rhythm generator (RG) and pattern formation (PF) networks [BUR 01, LAF 05, RYB 06a, RYB 06b]. The RG network generates the basic rhythm and alters it by producing phase shifts and rhythm resetting in response to sensory afferents and perturbations. The PF

network shapes the rhythm into spatiotemporal patterns of motor commands. The CPG separately controls the locomotor rhythm and motor commands in the RG and PF networks, respectively.

In this study, we develop a locomotion control system based on the two-layer hierarchical network model composed of the RG and PF networks. For the RG model, we produce the rhythm information for locomotor behavior using nonlinear oscillators and regulate the rhythm information based on phase resetting in response to touch sensor signals. For the PF model, we generate motor torques based on the rhythm information from the RG model to produce the joint movements. The following sections explain this locomotion control system.

3.2.2. Rhythm generator model using nonlinear oscillators

We construct the RG model to create rhythm information for the locomotor behavior through interactions of the robot mechanical system, the oscillator control system, and the environment. For this purpose, we use nonlinear oscillators to produce the basic locomotor rhythm, which is modulated by the phase resetting mechanism based on touch sensor signals.

We use an oscillator for the whole body, for each limb, or for each joint depending on the research objective. When we use an oscillator for each limb, the oscillators can manipulate the interlimb coordination pattern. When we use an oscillator for each joint, the oscillators can control the intralimb (intersegmental) coordination pattern, as well as the interlimb coordination pattern. In this study, we use an oscillator for the whole body for a simple biped robot model in section 3.3 and use an oscillator for each limb for actual biped robots in section 3.4.

When we use $N(\geq 1)$ oscillators (oscillator 1, $\dots$, N), we define ϕ_i ($i = 1, \dots, N$) as the phase of the oscillator i , and employ the following phase dynamics:

$$\dot{\phi}_i = \omega + g_{1i} + g_{2i}, \quad i = 1, \dots, N \quad [3.1]$$

where ω is the basic oscillator frequency that uses the same value for all the oscillators; g_{1i} ($i = 1, \dots, N$) is a function related to the interlimb and intralimb coordination (see section 3.2.4) and g_{2i} ($i = 1, \dots, N$) is a function related to the phase and rhythm modulation based on the phase resetting mechanism in response to touch sensor signals (see section 3.2.5).

3.2.3. Pattern formation model to determine the global parameters of limb kinematics

Physiological evidence has shown that spinocerebellar neurons receive sensory signals from proprioceptors and cutaneous receptors and encode the global information of the limb kinematics, such as the length and orientation of the limb axis that connects from the root to the tip of the limb [BOS 01, POP 02, POP 03]. In this study, we construct the PF model to determine the desired limb kinematics based on the length and orientation of the limb axis from the oscillator state in the RG model and to produce motor torques for establishing the desired kinematics.

3.2.4. Phase regulation based on interlimb and intralimb coordination

For establishing stable bipedal locomotion, interlimb coordination is crucial. For example, both legs generally move in antiphase to prevent toppling over, both arms also move in antiphase, and one arm and the contralateral leg move in phase. When we use the oscillator i for the limb i and design the desired limb kinematics by the oscillator state, the interlimb coordination pattern is represented by the phase difference among the oscillators. Therefore, the function g_{1i} in [3.1] is given by:

$$g_{1i} = - \sum_j K_{ij} \sin(\phi_i - \phi_j - \Delta_{ij}), \quad i = 1, \dots, N \quad [3.2]$$

where Δ_{ij} is the desired phase relation between the oscillators i and j and K_{ij} is a gain constant.

When we use an oscillator for each joint, the function g_{1i} can be used to regulate the intralimb coordination in a similar manner.

3.2.5. Sensory regulation based on phase resetting

We modulate the locomotion rhythm and its phase based on the phase resetting mechanism in response to touch sensor signals to produce adaptive locomotion through dynamic interactions between the robot mechanical system, the oscillator control system, and the environment. The function g_{2i} in [3.1] corresponds to this modulation. When we use the oscillator i for the limb i and the limb i lands on the ground, the phase ϕ_i of the oscillator i is reset to ϕ_0 at the landing. Therefore, the function g_{2i} is written by:

$$g_{2i} = (\phi_0 - \phi_i) \delta(t - t_i^{\text{land}}), \quad i = 1, \dots, N \quad [3.3]$$

where t_i^{land} is the time when the limb i lands on the ground and $\delta(\cdot)$ denotes Dirac's delta function.

3.3. Stability analysis using a simple biped robot model

In this section, we show how the locomotion control system contributes to increasing stability and robustness for locomotion using a compass model as a simple biped robot model [AOI 06b, AOI 07b].

3.3.1. Compass model

Figure 3.1 shows the compass model composed of a body and two legs that are connected at the hip. We assume that masses are concentrated at the hip and the tips of the legs. The body mass is M , the leg mass is m and the leg length is l . This model is constrained on the $x - y$ plane and walks to the x -direction. The tip of the stance leg is constrained on the ground and the stance leg can only rotate around the tip. This model has two degrees of freedom, θ_1 and θ_2 . θ_1 is the angle of the stance leg relative to the perpendicular line to the ground and is not directly controlled. On the other hand, θ_2 is the angle of the swing leg relative to the stance leg and is directly controlled by the input torque u . Acceleration due to gravity is g .

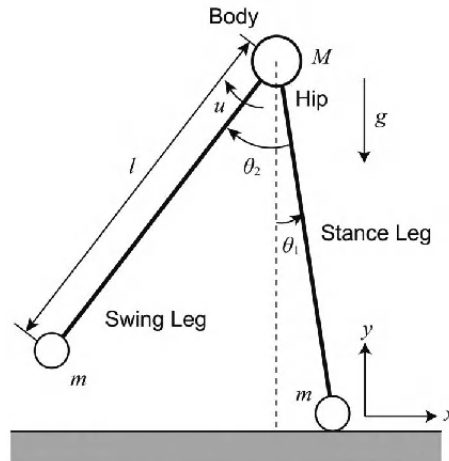


Figure 3.1. Compass model

The step cycle of the walking motion consists of two types of successive phases: swing and foot contact phases. In the following sections, we explain the governing equations for these phases.

3.3.1.1. *Swing phase model*

During the single-supported phase where the model is supported only by the stance leg and the swing leg is not in contact with the ground, the non-dimensional equation of motion is given by:

$$\begin{bmatrix} 1 + 2\beta(1 - \cos \theta_2) & -\beta(1 - \cos \theta_2) \\ -(1 - \cos \theta_2) & 1 \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} -\beta \sin \theta_2 (\dot{\theta}_2^2 - 2\dot{\theta}_1 \dot{\theta}_2) \\ -\dot{\theta}_1^2 \sin \theta_2 \end{bmatrix} + \begin{bmatrix} \beta \sin(\theta_1 - \theta_2) - \beta \sin \theta_1 - \sin \theta_1 \\ -\sin(\theta_1 - \theta_2) \end{bmatrix} = \begin{bmatrix} 0 \\ u' \end{bmatrix} \quad [3.4]$$

where $\beta = m/M$, $\tau = \sqrt{g/lt}$, $u' = u/mgl$ and $\dot{x}$ indicates the derivative of x with respect to τ .

3.3.1.2. *Foot contact model*

When the leg touches the ground, the leg tip receives an impulsive force from the ground. For the foot contact model, we assume that the leg tip has no slip and no rebound at the foot contact and the double-supported phase duration is sufficiently short relative to the step cycle. That is, immediately following the foot contact, the tip of the swing leg is in turn constrained on the ground and the stance leg leaves the ground. In other words, the swing leg instantaneously becomes the stance leg, and vice versa, immediately after the foot contact.

The geometric condition the foot contact (Figure 3.2) is given by:

$$2\theta_1 - \theta_2 = 0 \quad [3.5]$$

Because the roles of the legs switch between the swing and stance legs just following the foot contact, the relationship of the angles between immediately before and after the foot contact is given by:

$$\begin{bmatrix} \theta_1^+ \\ \theta_2^+ \end{bmatrix} = \begin{bmatrix} -\theta_1^- \\ -\theta_2^- \end{bmatrix} \quad [3.6]$$

where $()^-$ indicates the state just before the foot contact and $()^+$ indicates the state just after the foot contact.

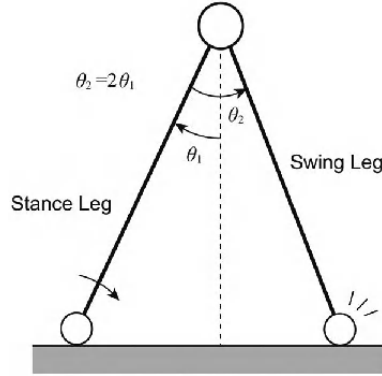


Figure 3.2. *Geometric condition for foot contact*

When the swing leg touches the ground, the leg tip receives an impact from the ground and the angular velocities suddenly change. We assume that the stance leg leaves the ground without interaction. From these assumptions, the angular velocities just after the foot contact are given by:

$$\begin{bmatrix} \dot{\theta}_1^+ \\ \dot{\theta}_2^+ \end{bmatrix} = \begin{bmatrix} \frac{2 \cos 2\theta_1^-}{2 + \beta(1 - \cos 4\theta_1^-)} \dot{\theta}_1^- \\ \frac{2 \cos 2\theta_1^- (1 - \cos 2\theta_1^-)}{2 + \beta(1 - \cos 4\theta_1^-)} \dot{\theta}_1^- \end{bmatrix} \quad [3.7]$$

3.3.2. Locomotion control system

We show how our locomotion control system is used for this compass model.

3.3.2.1. Rhythm generator model

Although the interlimb coordination plays important roles in achieving adaptive walking, we use only one oscillator for the RG model of the locomotion control system to keep the antiphase movements of the legs. Instead, we employ the amplitude of the oscillator, as well as the phase. We denote the amplitude and the phase of the oscillator by γ and ϕ , respectively. We use the following dynamics for the oscillator:

$$\begin{aligned} \dot{\gamma} &= 0 \\ \dot{\phi} &= \omega \end{aligned} \quad [3.8]$$

Note that because we use only one oscillator, the function g_{1i} in [3.1] vanishes. The function g_{2i} in [3.1] is incorporated in section 3.3.2.3.

3.3.2.2. Pattern formation model

In the PF model of the locomotion control system, we construct the desired joint motion from the oscillator state and produce motor torque for establishing the desired joint motion. Because the compass model does not change the leg length, we design the desired joint motion based on the orientation of the limb axis and use a simple oscillation by:

$$\theta_{2d} = \gamma \cos \phi + \gamma - s \quad [3.9]$$

where s is a parameter that determines the stride angle. When we do not use phase resetting, we set $\gamma = s$.

To produce this desired joint angle, the input torque u' is given by:

$$u' = -\kappa(\theta_2 - \theta_{2d}(\gamma, \phi)) - \sigma \dot{\theta}_2 \quad [3.10]$$

where κ and σ are gain constants.

For the foot contact model, we switched the leg roles between swing and stance legs in [3.6]. We also need to switch the desired joint angle in [3.9] at the foot contact. When we do not use phase resetting, this is satisfied by the following conditions for the oscillator state:

$$\begin{bmatrix} \gamma^+ \\ \phi^+ \end{bmatrix} = \begin{bmatrix} \gamma^- \\ \phi^- - \pi \end{bmatrix} \quad [3.11]$$

3.3.2.3. Phase resetting model

To incorporate the phase resetting mechanism, we reset the oscillator phase to ϕ_0 just after the foot contact. Instead of the function g_{2i} in [3.1], this is given by:

$$\phi^+ = \phi_0 \quad [3.12]$$

Accompanied by this phase resetting, we modulate the oscillator amplitude to avoid inducing discrete change in the desired joint angle by:

$$\gamma^+ = \frac{2s - \gamma^-(1 + \cos \phi^-)}{1 + \cos \phi_0} \quad [3.13]$$

Therefore, when we use phase resetting, we use [3.12] and [3.13] instead of [3.11].

3.3.3. Periodic solution of the walking motion

3.3.3.1. Assumption for the controlled joint angle

In this compass model, the joint angle θ_2 is controlled by a sufficient high-gain feedback control torque in [3.10]. We assume that the angle θ_2 is identical to the desired angle θ_{2d} during locomotion, that is,

$$\theta_2 = \theta_{2d}(\gamma, \phi) \quad [3.14]$$

Under this assumption, we can denote the state variables as $q^T = [\theta_1, \dot{\theta}_1, \gamma, \phi]$ and the equations of motion, the foot contact condition and the transition rule of the state variables from just before to just after a foot contact are summarized by:

$$\begin{cases} \dot{q} = f(q), & q^- \notin S_c \\ q^+ = h(q^-), & q^- \in S_c \end{cases} \quad [3.15]$$

where $S_c = \{q \mid r(q) = 2\theta_1 - \theta_{2d}(\gamma, \phi) = 0\}$ and

$$f(q) = \begin{bmatrix} f_1(q) \\ f_2(q) \\ f_3(q) \\ f_4(q) \end{bmatrix} = \begin{bmatrix} \dot{\theta}_1 \\ f_2(q) \\ 0 \\ \omega \end{bmatrix}$$

$$f_2(q) = \{\beta(1 - \cos \theta_{2d})\ddot{\theta}_{2d} + \beta \sin \theta_{2d}(\dot{\theta}_{2d}^2 - 2\dot{\theta}_1\dot{\theta}_{2d}) - \beta \sin(\theta_1 - \theta_{2d}) + \beta \sin \theta_1 + \sin \theta_1\} / \{1 + 2\beta(1 - \cos \theta_{2d})\}$$

$$h(q) = \begin{bmatrix} h_1(q) \\ h_2(q) \\ h_3(q) \\ h_4(q) \end{bmatrix} = \begin{bmatrix} -\theta_1 \\ \frac{2 \cos 2\theta_1}{2 + \beta(1 - \cos 4\theta_1)} \dot{\theta}_1 \\ h_3(q) \\ h_4(q) \end{bmatrix}$$

$$h_3(q) = \begin{cases} \gamma & \text{without resetting} \\ \frac{2s - \gamma(1 + \cos \phi)}{1 + \cos \phi_0} & \text{with resetting} \end{cases}$$

$$h_4(q) = \begin{cases} \phi - \pi & \text{without resetting} \\ \phi_0 & \text{with resetting} \end{cases}$$

3.3.3.2. Approximate periodic solution using perturbation method

Because the above equations can not be solved directly due to the strong nonlinearity, we attempt to solve them approximately and obtain the approximate periodic solution from just after a foot contact to just before the next foot contact. In particular, we consider the walking motion in which the tipping motion from the perpendicular line to the ground is small. In this case, the angles θ_1 and θ_{2d} are small and hence we define $s = \varepsilon$ and $\gamma = \varepsilon\Gamma$ ($\Gamma = 1$ for the case without phase resetting), where ε is a small parameter. We use power-series expansions for the state variables θ_1 and ϕ by:

$$\begin{aligned}\theta_1(\tau) &= \varepsilon X_0(\tau) + \varepsilon^3 X_1(\tau) + \cdots \\ \phi(\tau) &= (\omega\tau + \Phi_0) + \varepsilon^2 \Phi_1 + \cdots, \quad 0 \leq \tau \leq T\end{aligned}\quad [3.16]$$

where $T (= O(1))$ is the step period, that is the time interval from just after a foot contact to just before the next foot contact, and $\tau = 0$ and T indicates the time just after a foot contact and just before the next foot contact, respectively. These expansions yield:

$$\theta_{2d}(\gamma, \phi) = \varepsilon Y_0(\tau) + \varepsilon^3 Y_1(\tau) + \cdots, \quad 0 \leq \tau \leq T \quad [3.17]$$

where

$$\begin{cases} Y_0(\tau) = \Gamma \cos(\omega\tau + \Phi_0) + \Gamma - 1 \\ Y_1(\tau) = -\Gamma \sin(\omega\tau + \Phi_0)\Phi_1 \\ \vdots \end{cases}$$

By substituting [3.16] and [3.17] into the equation of motion in [3.15], we obtain:

$$\begin{aligned}\varepsilon[\ddot{X}_0 - \beta Y_0 - X_0] + \varepsilon^3\left[\ddot{X}_1 - \beta Y_1 - X_1 + \beta Y_0^2 \ddot{X}_0 - \frac{\beta}{2} Y_0^2 \ddot{Y}_0 + 2\beta \dot{X}_0 \dot{Y}_0 Y_0 \right. \\ \left. - \beta \dot{Y}_0^2 Y_0 + \frac{\beta}{2} X_0^2 Y_0 - \frac{\beta}{2} X_0 Y_0^2 + \frac{\beta}{6} Y_0^3 + \frac{1}{6} X_0^3\right] + O(\varepsilon^5) = 0\end{aligned}\quad [3.18]$$

where $X_i = X_i(\tau)$ and $Y_i = Y_i(\tau)$ ($i = 0, 1$). The substitution of the step period T and the state variables in [3.16] and [3.17] into the foot contact condition and transition rule in [3.15] gives the boundary condition and step

period for the approximate periodic solution by:

$$\begin{aligned}
\varepsilon \left[X_0(T) + \frac{1}{2} Y_0(0) \right] + \varepsilon^3 \left[X_1(T) + \frac{1}{2} Y_1(0) \right] + O(\varepsilon^5) &= 0 \\
\varepsilon \left[X_0(0) - \frac{1}{2} Y_0(0) \right] + \varepsilon^3 \left[X_1(0) - \frac{1}{2} Y_1(0) \right] + O(\varepsilon^5) &= 0 \\
\varepsilon \left[\dot{X}_0(0) - \dot{X}_0(T) \right] + \varepsilon^3 \left[\dot{X}_1(0) + 4\beta X_0^2(T) \dot{X}_0(0) \right. \\
&\quad \left. - \dot{X}_1(T) + 2X_0^2(T) \dot{X}_0(T) \right] + O(\varepsilon^5) = 0 \\
\varepsilon \left[Y_0(0) + Y_0(T) \right] + \varepsilon^3 \left[Y_1(0) + Y_1(T) \right] + O(\varepsilon^5) &= 0 \\
T = \begin{cases} \frac{\pi}{\omega} & \text{without resetting} \\ \frac{\pi - 2\phi_0}{\omega} & \text{with resetting} \end{cases} & \quad [3.19]
\end{aligned}$$

From [3.18] and [3.19], the equations for $X_0(\tau)$, Φ_0 and Γ are given by:

$$\begin{aligned}
\ddot{X}_0(\tau) - \beta Y_0(\tau) - X_0(\tau) &= 0 \\
\begin{cases} X_0(T) = -\frac{1}{2} Y_0(0) \\ X_0(0) = \frac{1}{2} Y_0(0) \\ \dot{X}_0(0) = \dot{X}_0(T) \\ Y_0(0) = -Y_0(T) \end{cases} & \quad [3.20]
\end{aligned}$$

The solution is given by:

$$\begin{aligned}
X_0(\tau) &= \left(\frac{\beta}{\omega^2 + 1} + \frac{1}{2} \right) \left(\frac{e^\tau}{1 - e^T} + \frac{e^{-\tau}}{1 - e^{-T}} \right) \cos \Psi - \frac{\beta}{\omega^2 + 1} \cos(\omega\tau + \Psi) \\
Y_0(\tau) &= \cos(\omega\tau + \Psi) \\
\Psi &= \begin{cases} 0 & \text{without resetting} \\ \phi_0 & \text{with resetting} \end{cases} \quad [3.21]
\end{aligned}$$

($\Phi_0 = 0$ for the case without phase resetting, $\Phi_0 = \phi_0$ and $\Gamma = 1$ for the case with phase resetting). This solution corresponds to the solution when we linearize equation [3.15] with respect to the angles θ_1 and θ_{2d} . When $\phi_0 = 0$, the solutions without and with phase resetting become identical.

Next, from [3.18] and [3.19], the equations for $X_1(\tau)$ and Φ_1 are given by:

$$\begin{aligned} \ddot{X}_1 - \beta Y_1 - X_1 + \beta Y_0^2 \dot{X}_0 - \frac{\beta}{2} Y_0^2 \ddot{Y}_0 + 2\beta \dot{X}_0 \dot{Y}_0 Y_0 \\ - \beta \dot{Y}_0^2 Y_0 + \frac{\beta}{2} X_0^2 Y_0 - \frac{\beta}{2} X_0 Y_0^2 + \frac{\beta}{6} Y_0^3 + \frac{1}{6} X_0^3 = 0 \end{aligned}$$

$$\begin{cases} X_1(T) = -\frac{1}{2}Y_1(0) \\ X_1(0) = \frac{1}{2}Y_1(0) \\ \dot{X}_1(0) + 4\beta X_0^2(T)\dot{X}_0(0) = \dot{X}_1(T) - 2X_0^2(T)\dot{X}_0(T) \\ Y_1(0) = -Y_1(T) \end{cases} \quad [3.22]$$

The solution of Φ_1 becomes:

$$\Phi_1 = \begin{cases} \frac{e^T + 1}{e^T - 1} \frac{(1 + 2\beta)(1 + 2\beta + \omega^2)}{8\beta\omega} & \text{without resetting} \\ 0 & \text{with resetting} \end{cases} \quad [3.23]$$

The solution of $X_1(\tau)$ is so complicated that we do not display the details here. It concludes that the approximate periodic solution of the state variables θ_1 and ϕ and step period T becomes equivalent to:

$$\begin{aligned} \theta_1(\tau) = \varepsilon \left[\left(\frac{\beta}{\omega^2 + 1} + \frac{1}{2} \right) \left(\frac{e^\tau}{1 - e^T} + \frac{e^{-\tau}}{1 - e^{-T}} \right) \cos \Psi \right. \\ \left. - \frac{\beta}{\omega^2 + 1} \cos(\omega\tau + \Psi) \right] + \dots \\ \phi(\tau) = \begin{cases} \omega\tau + \varepsilon \frac{e^T + 1}{e^T - 1} \frac{(1 + 2\beta)(1 + 2\beta + \omega^2)}{8\beta\omega} + \dots & \text{without resetting} \\ \omega\tau + \phi_0 & \text{with resetting} \end{cases} \\ T = \frac{\pi - 2\Psi}{\omega} \\ \Psi = \begin{cases} 0 & \text{without resetting} \\ \phi_0 & \text{with resetting} \end{cases} \quad [3.24] \end{aligned}$$

Figures 3.3(a) and (b) show the periodic solution of this approximate analysis and computer simulation ($\beta = 0.2$, $\omega = 1.5$ rad and $\varepsilon = 6^\circ$) for without and with phase resetting, respectively. In the computer simulation, we do not assume that the controlled angle θ_2 is identical to the desired angle θ_{2d}

and we actually controlled the angle θ_2 by the high-gain feedback torque in [3.10]. These figures verify that the obtained approximate solution is close to the rigorous solution and that the approximate solution to $O(\varepsilon^3)$ is closer to the rigorous solution than that to $O(\varepsilon)$.

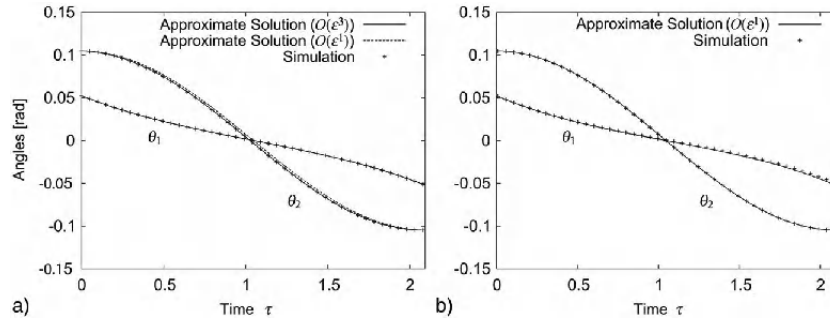


Figure 3.3. Periodic solution of approximate analysis and computer simulation ($\beta = 0.2$, $\omega = 1.5$ rad and $\varepsilon = 6^\circ$). a) Without phase resetting and b) with phase resetting ($\phi_0 = 0$)

3.3.4. Stability analysis

Whether the obtained periodic solution is stable depends on the gait parameters, such as β and ω . We conduct a stability analysis using a Poincaré map to investigate it.

A Poincaré map is a return map from one point on the Poincaré section to the next point on the Poincaré section. Periodic behavior results in a fixed point on the Poincaré section and stability is determined from the eigenvalues of the Jacobian matrix of the Poincaré map around the fixed point. Periodic motion is asymptotically stable if all of the eigenvalues are inside the unit circle on the complex plane, that is, all of the magnitudes of the eigenvalues are less than 1.

3.3.4.1. Jacobian matrix of the Poincaré map

In our stability analysis, we use the state just after the foot contact as the state on the Poincaré section. In our compass model, the walking motion is governed by continuous and discrete equations. By considering such a hybrid structure, the Jacobian matrix of the Poincaré map is given by the product of three matrices, B , D and E , induced by the disturbances in the discrete changes due to the foot contact, the change in its timing and the evolved perturbations during the continuous equation [COL 97].

To obtain the Jacobian matrix of the Poincaré map, we first define the periodic solution as $q^*(\tau)$, the step period as τ^* and the perturbed state from the periodic solution $q^*(\tau)$ from just after a foot contact to just before the next foot contact as $q^*(\tau) + \delta q(\tau)$. Then, the matrices B and D are given by:

$$\begin{aligned} B &= \partial_q h(q^*(\tau^*)) \\ D &= I - \frac{\dot{q}^*(\tau^*) \partial_q r(q^*(\tau^*))^T}{\partial_q r(q^*(\tau^*))^T \dot{q}^*(\tau^*)} \end{aligned} \quad [3.25]$$

where $\partial_q = \partial/\partial q$ and I is a unit matrix. The substitution of the perturbed state $q^*(\tau) + \delta q(\tau)$ into the equation of motion [3.15] gives:

$$\delta \dot{q}(\tau) = \partial_q f(q^*(\tau)) \delta q(\tau) \quad [3.26]$$

By integrating this equation from $\tau = 0$ to τ^* , the matrix E is derived from:

$$\delta q(\tau^*) = E \delta q(0) \quad [3.27]$$

The Jacobian matrix J of the Poincaré map is given by:

$$J = BDE \quad [3.28]$$

3.3.4.2. Stability characteristics of the periodic solution without phase resetting

When we do not use phase resetting, the amplitude of the oscillator remains constant ($\gamma = s$) during locomotion. Therefore, we use $q^T = [\theta_1, \dot{\theta}_1, \phi]$ for the state variables to investigate stability of the periodic solution.

When we use the approximate solution in [3.24], the matrices B , D and E are given by the power series of parameter ε^2 :

$$\begin{aligned} B &= B_0 + \varepsilon^2 B_1 + O(\varepsilon^4) \\ D &= D_0 + \varepsilon^2 D_1 + O(\varepsilon^4) \\ E &= E_0 + \varepsilon^2 E_1 + O(\varepsilon^4) \end{aligned} \quad [3.29]$$

This results in:

$$J = J_0 + \varepsilon^2 J_1 + O(\varepsilon^4) \quad [3.30]$$

where

$$\begin{aligned} J_0 &= B_0 D_0 E_0 \\ J_1 &= B_1 D_0 E_0 + B_0 D_1 E_0 + B_0 D_0 E_1 \end{aligned}$$

The substitution of the approximate periodic solution gives:

$$J_0 = \begin{bmatrix} 0 & 0 & 0 \\ * & 1 - \beta\omega^2 d(\beta, \omega) & -\frac{\beta\omega}{1 + \omega^2} \{2 - \beta\omega^2 d(\beta, \omega)\} \\ * & \omega(1 + \omega^2) d(\beta, \omega) & 1 - \beta\omega^2 d(\beta, \omega) \end{bmatrix} \quad [3.31]$$

where

$$d(\beta, \omega) = \frac{e^{\pi/\omega} + e^{-\pi/\omega} - 2}{1 + 2\beta + \omega^2} > 0$$

and J_1 is so complicated that we do not display here.

First, we investigate the stability of the periodic solution based on the matrix J_0 obtained from the solutions $X_0(\tau)$ and Φ_0 . That is, we employ only the first-order expansion of the state variables to approximately achieve the eigenvalues of the Jacobian matrix. The matrix J_0 has one zero eigenvalue ($\lambda_3 = 0$) and the other two eigenvalues $\lambda_{1,2}$ are given from the quadratic equation:

$$\lambda^2 - 2\{1 - \beta\omega^2 d(\beta, \omega)\}\lambda + 1 = 0 \quad [3.32]$$

Therefore, the eigenvalues $\lambda_{1,2}$ are given by:

$$\lambda_{1,2} = 1 - \beta\omega^2 d(\beta, \omega) \pm \sqrt{\{1 - \beta\omega^2 d(\beta, \omega)\}^2 - 1} \quad [3.33]$$

These eigenvalues are categorized according to the parameters β and ω as follows:

Case 1: $\lambda_{1,2}$ are complex conjugate

In this case, it follows that $0 < \beta\omega^2 d(\beta, \omega) < 2$ and the periodic solution is marginally stable from the following:

$$\lambda_1 \lambda_2 = \lambda \bar{\lambda} = |\lambda|^2 = 1 \quad [3.34]$$

Case 2: $\lambda_{1,2}$ are real and distinct

In this case, it follows that $\beta\omega^2 d(\beta, \omega) > 2$ and the periodic solution is unstable because there are stable and unstable eigenvalues from the following:

$$\begin{aligned}\lambda^2 - 1 &= 2\{1 - \beta\omega^2 d(\beta, \omega)\}\lambda - 2 \\ &= 2\sqrt{\{1 - \beta\omega^2 d(\beta, \omega)\}^2 - 1} \\ &\quad \times \left\{ \sqrt{[1 - \beta\omega^2 d(\beta, \omega)]^2 - 1} \pm [1 - \beta\omega^2 d(\beta, \omega)] \right\} \gtrless 0\end{aligned}\quad [3.35]$$

Case 3: $\lambda_{1,2}$ are degenerate

In this case, it follows that $\beta\omega^2 d(\beta, \omega) = 0, 2$ and the periodic solution is marginally stable from the following:

$$\lambda_{1,2} = 1, -1 \quad [3.36]$$

This linear stability analysis concludes that the periodic solution is marginally stable for $\beta\omega^2 d(\beta, \omega) \leq 2$ and otherwise the periodic solution is unstable.

To investigate the relationship of the stability between the parameters β and ω , we define $g_0(\beta, \omega) = \beta\omega^2 d(\beta, \omega) - 2$. It follows that:

$$\begin{aligned}\frac{\partial g_0}{\partial \omega}(\beta, \omega) &= -\frac{\beta(1 - e^{-\pi/\omega})}{(1 + 2\beta + \omega^2)^2} \left[\pi(1 + 2\beta + \omega^2)(1 + e^{\pi/\omega}) \right. \\ &\quad \left. + 2\omega(1 + 2\beta)(1 - e^{\pi/\omega}) \right] < 0 \\ \frac{\partial g_0}{\partial \beta}(\beta, \omega) &= \omega^2(1 + \omega^2) \frac{e^{\pi/\omega} + e^{-\pi/\omega} - 2}{(1 + 2\beta + \omega^2)^2} > 0\end{aligned}\quad [3.37]$$

In addition, it follows that:

$$\begin{aligned}\lim_{\omega \rightarrow +0} g_0(\beta, \omega) &= \infty \\ \lim_{\omega \rightarrow +\infty} g_0(\beta, \omega) &= -2 \\ \lim_{\beta \rightarrow +0} g_0(\beta, \omega) &= -2 \\ \lim_{\beta \rightarrow +\infty} g_0(\beta, \omega) &= \frac{1}{2} \left[\pi^2 - 4 + \sum_{n=2}^{\infty} \frac{2\omega^2}{(2n)!} \left(\frac{\pi}{\omega} \right)^{2n} \right] > 0\end{aligned}\quad [3.38]$$

From [3.37], $g_0(\beta, \omega)$ is monotonically decreasing and increasing with respect to the parameters β and ω , respectively. Therefore, from [3.38], the parameter ω (>0) that satisfies $g_0(\beta, \omega) = 0$ exists and is unique for each parameter β (>0) and also the parameter β (>0) that satisfies $g_0(\beta, \omega) = 0$ exists and is unique for each parameter ω (>0). From the implicit function theorem, we can define functions $\omega = \omega_0(\beta)$ and $\beta = \omega_0^{-1}(\omega)$ such that $g_0(\beta, \omega_0(\beta)) = g_0(\omega_0^{-1}(\omega), \omega) = 0$. Thus, it concludes that the periodic solution is marginally stable for $\omega \geq \omega_0(\beta)$ or, equivalently, $\beta \leq \omega_0^{-1}(\omega)$. Figure 3.4 shows the function $\omega = \omega_0(\beta)$. This corresponds to the boundary between the marginally stable and unstable region for the parameters β and ω .

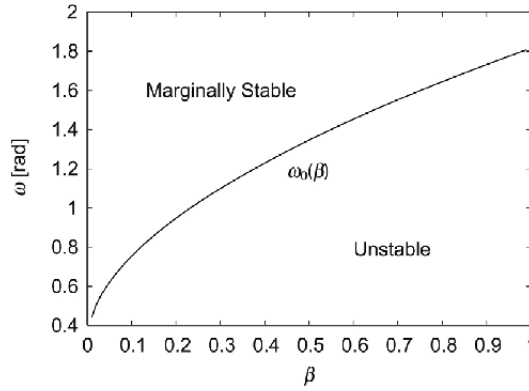


Figure 3.4. Function $\omega = \omega_0(\beta)$. This is the boundary between the marginally stable and unstable regions with respect to the parameters β and ω

This linear stability analysis reveals that the periodic solution is marginally stable or unstable depending on the parameters β and ω . The marginal stability result may be applicable in the limit when $\varepsilon = 0$ and the tipping motion and the stride of the walking motion disappear. However, marginal stability is a fragile feature and this analysis is inconclusive and insufficient to clarify the actual stability characteristics when $\varepsilon > 0$. Therefore, to examine the actual stability characteristics, we next achieve more rigorous eigenvalues of the Jacobian matrix by employing the second-order expansion of the state variables and matrix J_1 .

Figure 3.5 shows the contour of the maximum magnitude among three eigenvalues $\lambda_{1,2,3}$ of the matrix J to $O(\varepsilon^2)$ with respect to the parameters β and ω for $\varepsilon = 6^\circ$, where the thick line is the function $\omega = \omega_0(\beta)$. The maximum magnitude among the eigenvalues in the parameter region where we obtained the marginal stability in the first step of the stability analysis slightly decreases from 1, revealing that the periodic solution is asymptotically stable.

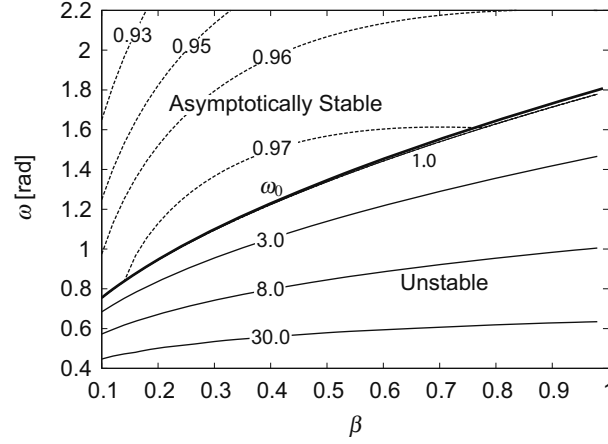


Figure 3.5. Contour of the maximum eigenvalue magnitude when we consider the higher order of the power-series expansions for the matrix J ($\varepsilon = 6^\circ$). Thick line shows the function $\omega = \omega_0(\beta)$

3.3.4.3. Stability improvement due to phase resetting

When we use phase resetting, the amplitude of the oscillator is modulated based on the foot contact event. Therefore, we use $q^T = [\theta_1, \dot{\theta}_1, \gamma, \phi]$ for the state variables to investigate stability of the periodic solution.

The substitution of the approximate periodic solution reveals that the matrix J_0 has two zero eigenvalues ($\lambda_{3,4} = 0$) and the other two eigenvalues $\lambda_{1,2}$ are obtained from the equation:

$$\lambda^2 - 2a_1\lambda + a_2 = 0 \quad [3.39]$$

where

$$\begin{aligned} a_1 &= \frac{1}{2\dot{X}_0^- - \dot{Y}_0^-} \left\{ -\ddot{X}_0^- E_- + \dot{X}_0^- \left(E_+ - \frac{1 - \cos \phi_0}{1 + \cos \phi_0} \right) \right. \\ &\quad \left. + \frac{\dot{Y}_0^-}{1 + \cos \phi_0} \left[\frac{\beta}{\omega^2 + 1} ((E_+ + 1) \cos \phi_0 - E_- \omega \sin \phi_0) + \beta(E_+ - 1) \right] \right\} \\ a_2 &= \frac{1}{2\dot{X}_0^- - \dot{Y}_0^-} \left\{ \dot{Y}_0^- + \frac{2\dot{Y}_0^-}{1 + \cos \phi_0} \left[\frac{\beta}{\omega^2 + 1} ((E_+ + 1) \cos \phi_0 \right. \right. \\ &\quad \left. \left. - E_- \omega \sin \phi_0) - \beta(E_+ - 1) \right] - 2 \frac{1 - \cos \phi_0}{1 + \cos \phi_0} (\dot{X}_0^- E_+ - \ddot{X}_0^- E_-) \right\} \end{aligned}$$

$$E_+ = \frac{e^T + e^{-T}}{2}, \quad E_- = \frac{e^T - e^{-T}}{2}$$

$$\dot{X}_0^- = \dot{X}_0(T), \quad \ddot{X}_0^- = \ddot{X}_0(T), \quad \dot{Y}_0^- = \dot{Y}_0(T)$$

The eigenvalues $\lambda_{1,2}$ are obtained by the function of the parameters β , ω and ϕ_0 and given by:

$$\lambda_{1,2}(\omega, \beta, \phi_0) = a_1 \pm \sqrt{a_1^2 - a_2} \quad [3.40]$$

First, and in particular, we consider the case when $\phi_0 = 0$. In this case, [3.40] becomes equivalent to:

$$\lambda_{1,2}(\omega, \beta, 0) = 1 - \beta\omega^2 d(\beta, \omega), \quad 0 \quad [3.41]$$

These eigenvalues reveal that the periodic solution is asymptotically stable for $\beta\omega^2 d(\beta, \omega) < 2$ and the periodic solution is unstable for $\beta\omega^2 d(\beta, \omega) > 2$. Regarding the stability region of the parameters β and ω , this result is the same as the case in which we do not use phase resetting. However, the stability is improved. Figures 3.6(a) and (b) show the maximum magnitude of the eigenvalues for the parameters β and ω in the cases without and with phase resetting, respectively, revealing that the stability is improved by employing phase resetting.

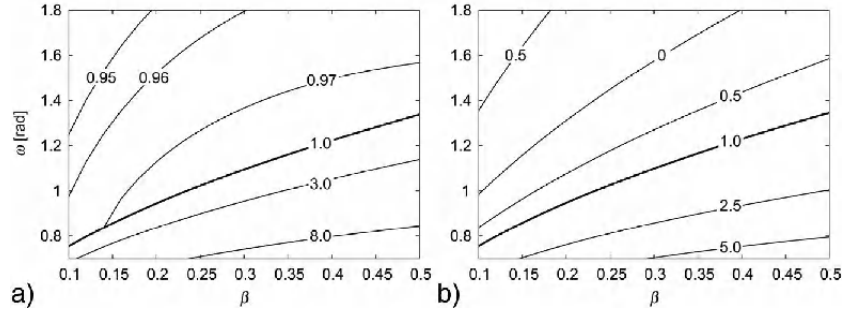


Figure 3.6. Contour of the maximum eigenvalue magnitude with respect to the parameters β and ω . a) Without phase resetting and b) with phase resetting ($\phi_0 = 0$)

Next, we consider the case that includes $\phi_0 \neq 0$. Figure 3.7 shows the transition of the stability region for the parameters β and ω due to the parameter ϕ_0 , and the comparison of the analytic and the simulation results ($\varepsilon = 6^\circ$).

This reveals that the asymptotic stability region is enlarged by increasing the parameter ϕ_0 .

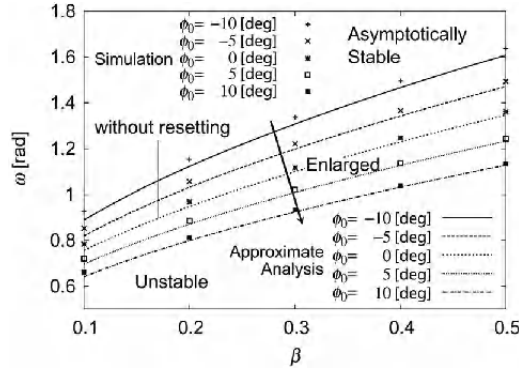


Figure 3.7. Transition of the stability for the parameter ϕ_0

When the parameter ϕ_0 exceeds a critical value, which depends on the parameters β and ω , the appearance of the stability region significantly changes. When the parameter ϕ_0 is less than the critical value, the decrease of β or/and the increase of ω increases the stability of the periodic motion and there is only one boundary with respect to the parameters β and ω , which divides into asymptotically stable and unstable regions as shown in Figure 3.7. On the other hand, when the parameter ϕ_0 is beyond the critical value, another boundary appears and the asymptotically stable region is formed between two unstable regions for the parameters β and ω , as shown in Figure 3.8. Therefore, in that case there is a region where the decrease of β or/and the increase of ω decreases the stability and thus it is necessary to use an adequate value for the parameter ϕ_0 .

We focus on the stability to achieve such parameter ϕ_0 . Figures 3.9(a) and (b) show the maximum magnitude of the eigenvalues versus the parameter ϕ_0 with respect to the parameters β and ω , respectively, revealing that it has an extreme value for the parameter ϕ_0 . Therefore, we use it as an optimal value of the parameter ϕ_0 . Such optimal value ϕ_0^* is obtained by the condition $a_1^2 - a_2 = 0$. Particularly when the optimal value ϕ_0^* is small, from the linear analysis it is given by:

$$\phi_0^* = \frac{(1 - \beta\omega^2d)^2}{\beta(1 - \beta\omega^2d) \left\{ \frac{16}{1 + 2\beta + \omega^2} - (\omega^2 - 4)d \right\} - 2} \quad [3.42]$$

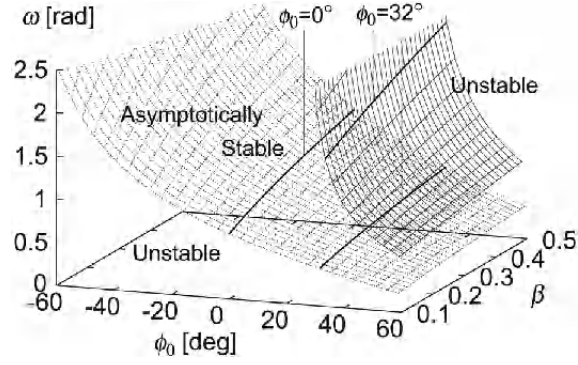


Figure 3.8. Stability region for the parameters with phase resetting

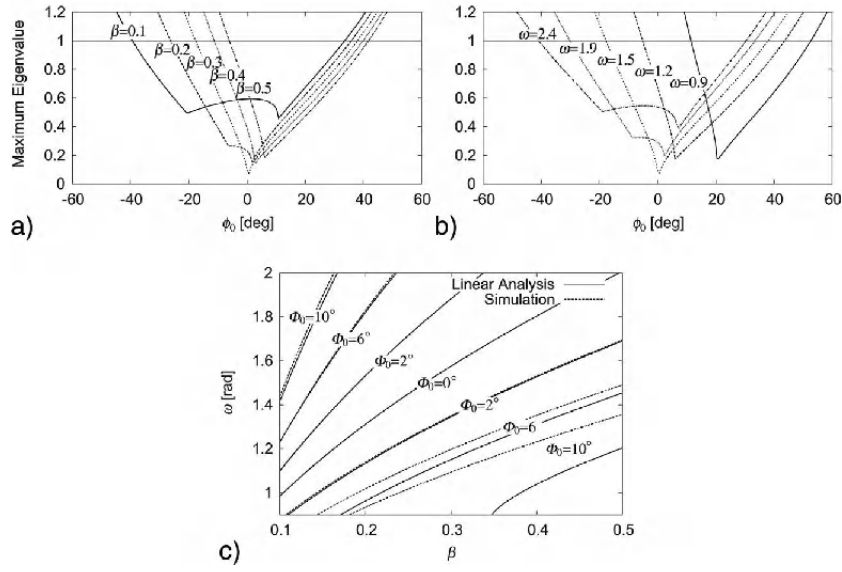


Figure 3.9. Maximum magnitude of the eigenvalues for the parameter ϕ_0 a) for the parameter β ($\omega = 1.5$ rad) and b) for the parameter ω ($\beta = 0.2$)). c) Optimal value ϕ_0^* for the parameters β and ω

Figure 3.9(c) shows the optimal value ϕ_0^* for the parameters β and ω obtained by the linear analysis and the rigorous computer simulation. This analysis concludes that resetting the phase of the oscillator into an adequate value and modifying the walking motion due to phase resetting based on the touch sensor signals increases the stability of the walking motion.

An analytical approach to show the improvement of stability and robustness of locomotion due to phase resetting is also conducted using a more plausible five-link biped robot model, which has torso and knee joints [AOI 11b]. In addition, the compass model used in this study shows period-doubling bifurcations leading to chaos by the parameters β and ω , similar to that observed in passive dynamic walking [GAR 98, GOS 98], where approximate analysis using perturbation method is useful to clarify stability characteristics inherent in locomotion dynamics [AOI 06a].

3.4. Experiment using biped robots

In this section, we show how the locomotion control system produces adaptive walking through the experiments using actual biped robots.

3.4.1. Locomotion control system

First, we show how our locomotion control system is used for actual biped robots.

3.4.1.1. Rhythm generator model

We denote the left and right legs by leg 1 and leg 2, respectively, and also denote the left and right arms by arm 1 and arm 2, respectively. For the RG model, we use six phase oscillators (leg 1, leg 2, arm 1, arm 2, trunk and inter oscillators). We define ϕ_L^i , ϕ_A^i , ϕ_T and ϕ_I ($i = 1, 2$) as the phases of the leg i , arm i , trunk and inter oscillators, respectively, and use the following dynamics for the phase dynamics [3.1]:

$$\begin{aligned}\dot{\phi}_I &= \omega + g_{1I} \\ \dot{\phi}_T &= \omega + g_{1T} \\ \dot{\phi}_A^i &= \omega + g_{1A}^i + g_{2A}^i \quad i = 1, 2 \\ \dot{\phi}_L^i &= \omega + g_{1L}^i + g_{2L}^i \quad i = 1, 2\end{aligned}\tag{3.43}$$

where g_{1L} , g_{1T} , g_{1A}^i and g_{1L}^i ($i = 1, 2$) are functions related to the interlimb coordination (see section 3.4.1.3) and g_{2A}^i and g_{2L}^i ($i = 1, 2$) are functions related to phase resetting (see section 3.4.1.4).

3.4.1.2. Pattern formation model

In the PF model, we use the simple leg kinematics consisting of the swing and stance phases in reference to the length and orientation of the limb axis in the pitch plane (Figure 3.10). For the swing phase, the ankle pitch joint follows the simple closed curve, which includes an anterior extreme position (AEP) and a posterior extreme position (PEP). It starts from the PEP and continues until the foot touches the ground. During the stance phase, it traces out a straight line from the landing position (LP) to the PEP. In both the swing and stance phases, the angular movement of the ankle pitch joint is designed so that the foot is parallel to the line that connects points AEP and PEP.

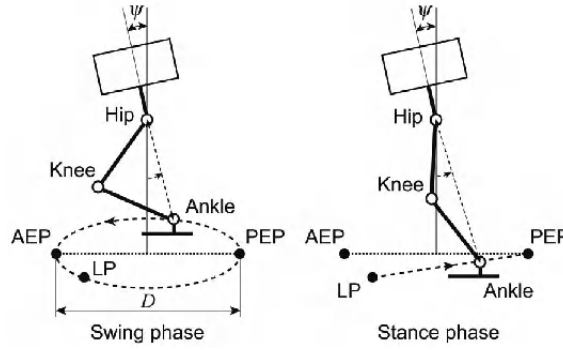


Figure 3.10. Desired leg kinematics composed of the swing and the stance phases. When the foot lands on the ground, the trajectory changes from the swing to the stance phase. When the foot reaches the posterior extreme position (PEP), the trajectory moves into the swing phase

We denote D as the distance between the AEP and PEP. We denote the swing and stance phase durations by T_{sw} and T_{st} , respectively, for the case that the foot touches the ground at the AEP ($LP = AEP$). The nominal duty factor β , which is the ratio between the stance phase and the gait cycle, the basic frequency ω , the stride length S , and the locomotion speed v are then given by:

$$\beta = \frac{T_{st}}{T_{sw} + T_{st}}$$

$$\omega = \frac{2\pi}{T_{sw} + T_{st}}$$

$$\begin{aligned}
S &= \frac{T_{\text{sw}} + T_{\text{st}}}{T_{\text{st}}} D \\
v &= \frac{D}{T_{\text{st}}}
\end{aligned} \tag{3.44}$$

The trunk is at the angle of ψ to the line perpendicular to the line connecting the AEP and the PEP. From the inverse kinematics, the two trajectories for the swing and stance phases provide the desired angles of the hip, knee and ankle pitch joints of the leg i by the function of the phase ϕ_L^i of the leg i oscillator, where we use $\phi_L^i = 0$ at the PEP and $\phi_L^i = \phi_{\text{AEP}} (= 2\pi(1 - \beta))$ at the AEP.

To increase the stability of bipedal locomotion in three-dimensional space, we also use the hip and ankle roll joints. We design the desired angles by $R \cos(\phi_T + \varphi)$ using the phase ϕ_T of the trunk oscillator, where R is the amplitude of the roll motion and φ determines the phase relationship between the leg movements in the pitch and roll planes.

In our study, we deal with both bipedal and quadrupedal locomotion of a biped robot. For quadrupedal locomotion, the arm motions are also designed from the inverse kinematics using the shoulder and elbow pitch joints in a similar manner to the pitch joints of the legs.

3.4.1.3. Interlimb coordination

For the interlimb coordination, we use the desired phase relations by $\phi_A^1 - \phi_A^2 = \pi$, $\phi_L^1 - \phi_L^2 = \pi$ and $\phi_A^1 - \phi_L^2 = 0$ ($\phi_A^2 - \phi_L^1 = 0$) so that both legs and arms generally move in antiphase, and one arm and the contralateral leg move in phase. The functions g_{1I} , g_{1T} , g_{1A}^i and g_{1L}^i in [3.43] are given as follows by using the phase differences between the oscillators based on the inter oscillator:

$$\begin{aligned}
g_{1I} &= - \sum_{i=1}^2 K_A \sin(\phi_I - \phi_A^i - (-1)^i \pi/2) \\
&\quad - \sum_{i=1}^2 K_L \sin(\phi_I - \phi_L^i + (-1)^i \pi/2) \\
g_{1T} &= -K_T \sin(\phi_T - \phi_I) \\
g_{1A}^i &= -K_A \sin(\phi_A^i - \phi_I + (-1)^i \pi/2) \quad i = 1, 2 \\
g_{1L}^i &= -K_L \sin(\phi_L^i - \phi_I - (-1)^i \pi/2) \quad i = 1, 2
\end{aligned} \tag{3.45}$$

where K_L , K_A and K_T are gain constants.

3.4.1.4. Phase resetting

As the sensory modulation based on phase resetting, when the foot of the leg i (the hand of the arm i) lands on the ground, the phase ϕ_L^i of the leg i oscillator (the phase ϕ_A^i of the arm i oscillator) is reset to ϕ_{AEP} at the landing ($i = 1, 2$). Therefore, the functions g_{2A}^i and g_{2L}^i in [3.43] are expressed as:

$$\begin{aligned} g_{2A}^i &= (\phi_{AEP} - \phi_A^i) \delta(t - t_{Aland}^i) \quad i = 1, 2 \\ g_{2L}^i &= (\phi_{AEP} - \phi_L^i) \delta(t - t_{Lland}^i) \quad i = 1, 2 \end{aligned} \quad [3.46]$$

where t_{Lland}^i (t_{Aland}^i) is the time when the foot of the leg i (the hand of the arm i) lands on the ground ($i = 1, 2$). Note that the touch sensor signals not only modulate the locomotor rhythm and its phase but also switch the desired motion from the swing to the stance phase, as described in section 3.4.1.2.

3.4.2. Experimental results

To show the usefulness of our locomotion control system in the real world, we conducted various experiments using actual biped robots.

3.4.2.1. Adaptive walking to environmental changes

In this study, we used a biped robot, HOAP-1 (Fujitsu Automation Ltd.). We investigated if our locomotion control system produces adaptive walking to environmental changes, by discontinuously changing the slope angle of the floor [AOI 05]. Specifically, at the beginning the robot walked on a level surface, then on the slope and finally again on the flat surface. We employed two types of slope; upslope and downslope.

When we did not use phase resetting, the robot easily fell over. However, the robot with phase resetting achieved stable walking, as shown in Figures 3.11(a) and (b). Figures 3.11(c) and (d) show the profiles of the step cycle for upslope and downslope, respectively, revealing that the robot walks adaptively by changing the step cycle depending on the slope angle of the floor due to phase resetting. This adaptability is not a characteristic that we specifically designed, but it emerges through dynamic interaction between the robot's mechanical system, the oscillator control system and the environment.

3.4.2.2. Adaptive turning walk

Next, we dealt with turning walk using HOAP-1, where we established the turning walk by integrating straight and curved walk [AOI 07a]. Specifically, at the beginning the robot walks straight and then begins to circle to the left.

Its motion returns to straight walking and changes to curved walking to the right. Finally, it walks straight again. By regulating the number of steps for each walking pattern, the robot walks in a figure of eight.

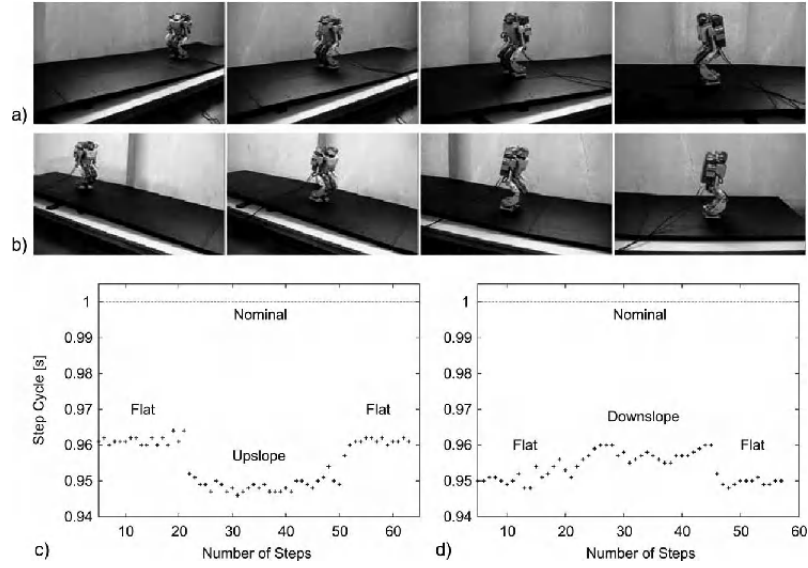


Figure 3.11. Experimental result for discontinuously change of the floor's slope, where at the beginning the robot walks on the flat surface, then on the slope, and finally again on the flat surface. a) Snapshots for upslope and b) for downslope. c) Step cycle versus number of steps for upslope and d) for downslope

Although the robot without phase resetting easily fell down, the robot with phase resetting achieved stable turning walk, as shown in Figure 3.12(a). Figure 3.12(b) shows the profile of the duty factors during the turning walk, revealing that the robot walks adaptively by changing the duty factors between straight and curved walk. Furthermore, the duty factors differ between the left and right legs for curved walk, similar to those observed in human curved walk [COU 03]. This adaptability also emerges through dynamic interaction between the robot mechanical system, the oscillator control system and the environment.

3.4.2.3. Adaptive splitbelt treadmill walking

To establish adaptive locomotion, interlimb coordination is an important factor. To investigate the mechanism controlling the interlimb coordination during walking, a splitbelt treadmill has been used [MOR 06, REI 05].

This treadmill is equipped with two parallel belts and the belt speeds can be controlled independently, which allows us to control the environmental situation to walk. We developed a biped robot and splitbelt treadmill for the robot and investigated the splitbelt treadmill walk of the robot [AOI 11a].

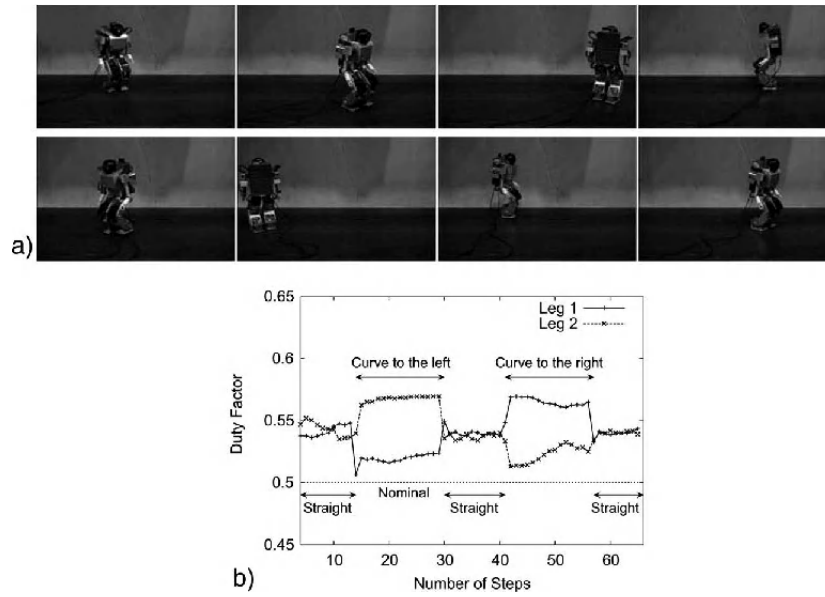


Figure 3.12. Experimental result for turning walk, where the robot walks in a figure eight. a) Snapshots and b) duty factors versus number of steps

When we did not use phase resetting, the robot easily fell down. However, the robot with phase resetting established stable walking despite a large discrepancy between the belt speeds. Instead, the relative phase between the leg movements shifted from antiphase and the duty factors of the legs varied depending on the speed discrepancy between the belts, similar to those observed in human splitbelt treadmill walking [MOR 06, REI 05].

3.4.2.4. Adaptive gait transition from quadrupedal to bipedal locomotion

Although many studies have investigated methods to achieve stable locomotor behaviors for various gaits, their transitions have not been thoroughly examined. In particular, the gait transition from quadrupedal to bipedal locomotion needs drastic changes in the robot posture and the reduction of the number of supporting limbs. Therefore, the stability greatly changes during the transition. To achieve such a gait transition, the following

two issues are crucial: (1) because a robot has many degrees of freedom, it is difficult to determine how to produce robot movements to connect one gait pattern to another, in other words, how to construct adequate constraint conditions in motion planning; (2) even if the robot establishes stable gait patterns, it may fall over during the gait transition and it is difficult to establish stable gait transition without falling over.

We applied our locomotion control system while incorporating the motion planning for the gait transition based on the physiological concept of kinematic synergy [ALE 98, FRE 06, FUN 10]. Similar to the above experiments, although the robot without phase resetting easily fell down, the robot with phase resetting achieved adaptive gait transition, as shown in Figure 3.13 [AOI 12].

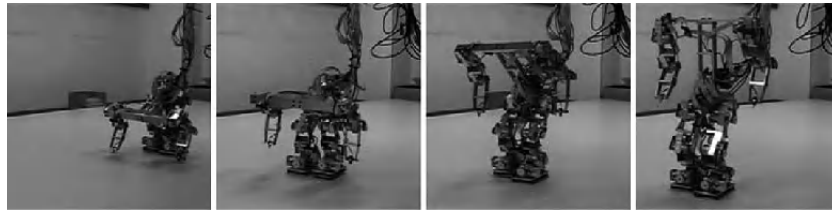


Figure 3.13. *Snapshots of the gait transition from quadrupedal to bipedal locomotion*

3.5. Conclusion

In this chapter, we designed a locomotion control system for a biped robot using nonlinear oscillators with phase resetting based on the physiological concept of CPG. We demonstrated the usefulness of this control system from the stability analysis using a simple biped robot model and from the experiments using actual biped robots. The stability analysis shows that the locomotion control system produces stable locomotor behavior and in addition shows that phase resetting improves stability for the gait parameters. The robot experiment shows that the locomotion control system using phase resetting produces adaptive walking of biped robots to environmental changes, such as discontinuous changes in slope angle and belt speeds of the splitbelt treadmill, and furthermore establishes adaptive turning walk and gait transition from quadrupedal to bipedal locomotion.

In locomotion of humans and animals, neuromechanical coordination is crucial. Although the phase resetting mechanism in biological systems remains unclear, we simply designed the sensory regulation model based

on phase resetting. Using phase oscillators allowed us to easily incorporate this to our locomotion control system. This modulates the locomotor rhythm depending on the foot contact information. In addition, this switches the desired kinematics from the swing phase to the stance phase. Because the swing and stance legs have different roles for locomotion dynamics, adequate timing to switch the desired kinematics is crucial. Despite this modulation being simple, it greatly contributes to generating various adaptive locomotor behaviors. In the future, we would like to employ a more sophisticated physical model and a biped robot to further clarify the stability mechanism and to construct a design principle to generate adaptive walking of legged robots.

3.6. Acknowledgments

This study is supported in part by a Grant-in-Aid for Scientific Research (B) No. 23360111 from the Ministry of Education, Culture, Sports, Science, and Technology of Japan, and by JST, CREST.

3.7. Bibliography

- [ALE 98] ALEXANDROV A., FROLOV A., MASSION J., “Axial synergies during human upper trunk bending”, *Experimental Brain Research*, vol. 118, pp. 210–220, 1998.
- [AOI 05] AOI S., TSUCHIYA K., “Locomotion control of a biped robot using nonlinear oscillators”, *Autonomous Robots*, vol. 19, no. 3, pp. 219–232, 2005.
- [AOI 06a] AOI S., TSUCHIYA K., “Bifurcation and chaos of a simple walking model driven by a rhythmic signal”, *International Journal of Non-Linear Mechanics*, vol. 41, no. 3, pp. 438–446, 2006.
- [AOI 06b] AOI S., TSUCHIYA K., “Stability analysis of a simple walking model driven by an oscillator with a phase reset using sensory feedback”, *IEEE Transactions on Robotics*, vol. 22, no. 2, pp. 391–397, 2006.
- [AOI 07a] AOI S., TSUCHIYA K., “Adaptive behavior in turning of an oscillator-driven biped robot”, *Autonomous Robots*, vol. 23, no. 1, pp. 37–57, 2007.
- [AOI 07b] AOI S., TSUCHIYA K., “Self-stability of a simple walking model driven by a rhythmic signal”, *Nonlinear Dynamics*, vol. 48, no. 1–2, pp. 1–16, 2007.
- [AOI 10] AOI S., OGIHARA N., FUNATO T., *et al.*, “Evaluating functional roles of phase resetting in generation of adaptive human bipedal walking with a physiologically based model of the spinal pattern generator”, *Biological Cybernetics*, vol. 102, no. 5, pp. 373–387, 2010.

- [AOI 11a] AOI S., FUJIKI S., YAMASHITA T., *et al.*, “Generation of adaptive splitbelt treadmill walking by a biped robot using nonlinear oscillators with phase resetting”, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2274–2279, 2011.
- [AOI 11b] AOI S., TSUCHIYA K., “Generation of bipedal walking through interactions among the robot dynamics, the oscillator dynamics, and the environment: stability characteristics of a five-link planar biped robot”, *Autonomous Robots*, vol. 30, no. 2, pp. 123–141, 2011.
- [AOI 12] AOI S., EGI Y., SUGIMOTO R., *et al.*, “Functional roles of phase resetting in the gait transition of a biped robot from quadrupedal to bipedal locomotion”, *IEEE Transactions on Robotics*, vol. 28, no. 6, pp. 1244–1259, 2012.
- [BOS 01] BOSCO G., POPPELE R., “Proprioception from a spinocerebellar perspective”, *Physiological Reviews*, vol. 81, pp. 539–568, 2001.
- [BUR 01] BURKE R., DEGTYARENKO A., SIMON E., “Patterns of locomotor drive to motoneurons and last-order interneurons: clues to the structure of the CPG”, *Journal of Neurophysiology*, vol. 86, pp. 447–462, 2001.
- [COL 97] COLEMAN M., CHATTERJEE A., RUINA A., “Motions of a rimless spoked wheel: a simple three-dimensional system with impacts”, *Dynamics and Stability of Systems*, vol. 12, no. 3, pp. 139–160, 1997.
- [CON 87] CONWAY B., HULTBORN H., KIEHN O., “Proprioceptive input resets central locomotor rhythm in the spinal cat”, *Experimental Brain Research*, vol. 68, pp. 643–656, 1987.
- [COU 03] COURTINE G., SCHIEPPATI M., “Human walking along a curved path. II. Gait features and EMG patterns”, *European Journal of Neuroscience*, vol. 18, no. 1, pp. 191–205, 2003.
- [DUY 77] DUYSSENS J., “Fluctuations in sensitivity to rhythm resetting effects during the cat’s step cycle”, *Brain Research*, vol. 133, no. 1, pp. 190–195, 1977.
- [DUY 00] DUYSSENS J., CLARAC F., CRUSE H., “Load-regulating mechanisms in gait and posture: comparative aspects”, *Physiological Reviews*, vol. 80, pp. 83–133, 2000.
- [FOR 73] FORSSBERG H., GRILLNER S., “The locomotion of the acute spinal cat injected with clonidine i.v.”, *Brain Research*, vol. 50, pp. 184–186, 1973.
- [FRE 06] FREITAS S., DUARTE M., LATASH M., “Two kinematic synergies in voluntary whole-body movements during standing”, *Journal of Neurophysiology*, vol. 95, pp. 636–645, 2006.
- [FUN 10] FUNATO T., AOI S., OSHIMA H., *et al.*, “Variant and invariant patterns embedded in human locomotion through whole body kinematic coordination”, *Experimental Brain Research*, vol. 205, no. 4, pp. 497–511, 2010.

- [GAR 98] GARCIA M., CHATTERJEE A., RUINA A., *et al.*, “The simplest walking model: stability, complexity, and scaling”, *ASME Journal of Biomechanical Engineering*, vol. 120, no. 2, pp. 281–288, 1998.
- [GOS 98] GOSWAMI A., THUÏLOT B., ESPIAU B., “A study of the passive gait of a compass-like biped robot: symmetry and chaos”, *International Journal of Robotics Research*, vol. 17, no. 12, pp. 1282–1301, 1998.
- [GRI 75] GRILLNER S., “Locomotion in vertebrates: central mechanisms and reflex interaction”, *Physiological Reviews*, vol. 55, no. 2, pp. 247–304, 1975.
- [GUE 95] GUERTIN P., ANGEL M., PERREAULT M.-C., *et al.*, “Ankle extensor group i afferents excite extensors throughout the hindlimb during fictive locomotion in the cat”, *Journal of Physiology*, vol. 487, no. 1, pp. 197–209, 1995.
- [IJS 07] IJSPEERT A., CRESPI A., RYCZKO D., *et al.*, “From swimming to walking with a salamander robot driven by a spinal cord model”, *Science*, vol. 315, pp. 1416–1420, 2007.
- [IJS 08] IJSPEERT A., “Central pattern generators for locomotion control in animals and robots: a review”, *Neural Networks*, vol. 21, no. 4, pp. 642–653, 2008.
- [KIM 07] KIMURA H., FUKUOKA Y., COHEN A., “Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts”, *International Journal of Robotics Research*, vol. 26, no. 5, pp. 475–490, 2007.
- [LAF 05] LAFRENIERE-ROULA M., MCCREA D., “Deletions of rhythmic motoneuron activity during fictive locomotion and scratch provide clues to the organization of the mammalian central pattern generator”, *Journal of Neurophysiology*, vol. 94, pp. 1120–1132, 2005.
- [LIU 08] LIU G., HABIB M., WATANABE K., *et al.*, “Central pattern generators based on Matsuoka oscillators for the locomotion of biped robots”, *Artificial Life and Robotics*, vol. 12, no. 1, pp. 264–269, 2008.
- [MOR 06] MORTON S., BASTIAN A., “Cerebellar contributions to locomotor adaptations during splitbelt treadmill walking”, *Journal of Neuroscience*, vol. 26, no. 36, pp. 9107–9116, 2006.
- [NAK 04] NAKANISHI J., MORIMOTO J., ENDO G., *et al.*, “Learning from demonstration and adaptation of biped locomotion”, *Robotics and Autonomous Systems*, vol. 47, no. 2-3, pp. 79–91, 2004.
- [NAK 06] NAKANISHI M., NOMURA T., SATO S., “Stumbling with optimal phase reset during gait can prevent a humanoid from falling”, *Biological Cybernetics*, vol. 95, pp. 503–515, 2006.
- [NOM 09] NOMURA T., KAWA K., SUZUKI Y., *et al.*, “Dynamic stability and phase resetting during biped gait”, *Chaos*, vol. 19, pp. 026103-1–026103-12, 2009.
- [ORL 99] ORLOVSKY G., DELIAGINA T., GRILLNER S., *Neuronal Control of Locomotion: From Mollusc to Man*, Oxford University Press, 1999.

- [POP 02] POPPELE R., BOSCO G., RANKIN A., “Independent representations of limb axis length and orientation in spinocerebellar response components”, *Journal of Neurophysiology*, vol. 87, pp. 409–422, 2002.
- [POP 03] POPPELE R., BOSCO G., “Sophisticated spinal contributions to motor control”, *Trends in Neurosciences*, vol. 26, pp. 269–276, 2003.
- [REI 05] REISMAN D., BLOCK H., BASTIAN A., “Interlimb coordination during locomotion: what can be adapted and stored?”, *Journal of Neurophysiology*, vol. 94, pp. 2403–2415, 2005.
- [RYB 06a] RYBAK I., SHEVTSOVA N., LAFRENIERE-ROULA M., *et al.*, “Modelling spinal circuitry involved in locomotor pattern generation: insights from deletions during fictive locomotion”, *Journal of Physiology*, vol. 577, no. 2, pp. 617–639, 2006.
- [RYB 06b] RYBAK I., STECINA K., SHEVTSOVA N., *et al.*, “Modelling spinal circuitry involved in locomotor pattern generation: insights from the effects of afferent stimulation”, *Journal of Physiology*, vol. 577, no. 2, pp. 641–658, 2006.
- [SCH 98] SCHOMBURG E., PETERSEN N., BARAJON I., *et al.*, “Flexor reflex afferents reset the step cycle during fictive locomotion in the cat”, *Experimental Brain Research*, vol. 122, no. 3, pp. 339–350, 1998.
- [SHI 76] SHIK M., ORLOVSKY G., “Neurophysiology of locomotor automatism”, *Physiological Reviews*, vol. 56, pp. 465–501, 1976.
- [STE 10] STEINGRUBE S., TIMME M., WÖRGÖTTER F., *et al.*, “Self-organized adaptation of a simple neural circuit enables complex robot behaviour”, *Nature Physics*, vol. 6, pp. 224–230, 2010.
- [YAK 04] YAKOVENKO S., GRITSENKO V., PROCHAZKA A., “Contribution of stretch reflexes to locomotor control: a modeling study”, *Biological Cybernetics*, vol. 90, pp. 146–155, 2004.
- [YAM 03a] YAMASAKI T., NOMURA T., SATO S., “Phase reset and dynamic stability during human gait”, *Biosystems*, vol. 71, pp. 221–232, 2003.
- [YAM 03b] YAMASAKI T., NOMURA T., SATO S., “Possible functional roles of phase resetting during walking”, *Biological Cybernetics*, vol. 88, pp. 468–496, 2003.

Chapter 4

Modeling a Human's Learning Processes toward Continuous Learning Support System

This chapter describes the learning support system for a human to achieve continuous learning. The objective of this research is to make a prototype system on the basis of a learning process model to guide a human to achieve continuous learning.

First, the need for continuous learning in mechatronics engineering is described, and then we give an introduction to research on learning support systems for a human. We point out the importance of support for awareness and understanding of learning processes. Second, requirements for modeling a learning process to achieve continuous learning are discussed. The main issue is “who generates a learning goal of a learner?” When the learner accomplishes the learning goal, he/she stops learning.

The main problem is how to keep supplying new goals to a learner for achieving continuous learning. A useful suggestion is from research in metalearning in education. To encourage the sense of continuous awareness towards goal discovery, we propose an idea to provide a human learner with invisible goals. Then, this chapter formalizes the continuous learning by a simple maze model with *invisible goals* and designs *the maze sweeping task*

that involves *multiple solutions* and *goals*. Under *the maze sweeping task* to collect as many optimal solutions as possible, we perform the experiment to make a comparison between *under invisible goals* condition and under visible goals condition. Experimental results showed that our continuous learning support system is effective, besides *the invisible goals* are more effective to assist a greater understanding in a learning process than the visible goals. Finally, several issues to develop our approach are discussed.

4.1. Introduction

4.1.1. *The need for continuous learning in mechatronics engineering*

In recent years, a new design approach for human and a mechatronics system that considers a human's intelligent behaviors such as skill level going up or learning has been studied [HAB 06, HAB 08, SUZ 10]. Human adaptive mechatronics (HAM) [SUZ 10] is the new mechatronics concept: the HAM system changes its mechanism and/or operating mode according to an operator's skill level to support the operator appropriately. By virtue of a human's adaptation ability, the operator may not only achieve his/her tasks but also up one's skill level.

To design HAM for humans to up their skill level, there are several research targets on humans' intelligent and mental processes such as individual learning, team-based learning or creative thinking [HAB 08]. This chapter focuses on the design method for an individual human's continuous learning support system toward open-ending learning.

4.1.2. *Learning support systems for a human*

Since the 1980s, computer systems have been used in many different ways to assist in human learning. Computer-based systems have been applied in the field of human learning for three different purposes [SKL 10]: (1) to replicate human behavior, (2) to model human behavior or (3) to augment human behavior. The goal of the first class of system is to approximate the outcomes of human behavior such as expert systems and some robotic systems. The goal of the second class of system is to imitate the processes underlying human behavior. The purpose behind modeling of humans is in order to better understand how and perhaps why humans act as they do. Research in this area comes under social science, particularly psychology, so

that there is no computer model. The goal of the third class of systems, such as an agent-based system, is to facilitate the acquisition of knowledge or help the learner to become proficient. The aim is to provide artificial partners for joint human activity. As discussed above, the position of our research is based on the second class toward the third class.

A learning process has been focused on several research areas, man-machine systems, education and management in business. The first learning process is operation learning in man-machine systems [RAU 96]. In that, to estimate the degree of the learning process level, the complexity of observed behaviors is calculated by one of the standard metrics in graph theory. The second learning process is computer-mediated education through a virtual learning environment or through online learning support such as e-learning. In that, the role of educators can change toward facilitating and guiding students instead of transmission of knowledge. Since it is important to support the awareness and understanding of learning processes, several visualization methods such as activity visualization [HAR 99] or visualizing the students' communication activities in a Web-based system [MAY 11] has been proposed. However, there is a basic problem in that these previous methods commonly depend on observable behaviors or activities. On the other hand, observing the learning process of a human is very difficult since it is a mental process. So, it is necessary to add a new criterion to observe the learning process of a human.

4.1.3. Modeling a learning process to achieve continuous learning

In the field of business management, psychological research on human motivation comes to the fore. Pink [PIN 09] examines the three elements of true motivation – autonomy, mastery and purpose. Since an organization needs to be successful in today's rapidly changing environment, there is a great need to facilitate continuous improvement and innovation in business processes. Therefore, a learning process model to achieve continuous improvement has been proposed [BUC 96]. In this section, we summarize the Buckler's learning process model.

A learning process is defined as a process that results in changed behavior [BUC 96]. There are three elements for this process to be effective: the hows, the whys and the whats of learning. The "how" of learning is a technique to help the learning process. Table 4.1 shows the summary of two kinds of major learning methods from [BUC 96]. Our approach is mainly based on the discovery model and it is a learner-centered approach. Our system empowers

a learner to achieve the goal, and to challenge under risk of failure. Thereby, the sense of autonomy enhances intrinsic motivation toward creativity.

Model	Taught model	Discovery model
Focus	Teacher-centered	Learner-centered
Motivation	Extrinsic	Intrinsic
Culture	Controlled	Empowered/Autonomous
Advantages	Consistency	Creativity
Disadvantages	Create barriers to change	Higher risk of failure

Table 4.1. Two kinds of major learning methods

The “whys” of learning creates an environment that provides meaning. The “whats” of learning enables a focus on goals or tasks. Figure 4.1 shows an overview of a learning process model for the positioning of the whats and whys of learning. These are the input of the learning process. The output of the learning process is a learning result, it is observed as a change of behavior of a learner.

The learning process consists of several stages. Table 4.2 shows the stages of the learning process with the role of leadership [BUC 96]. This chapter focuses on the stages of awareness, understanding and commitment as shown in Figure 4.1.

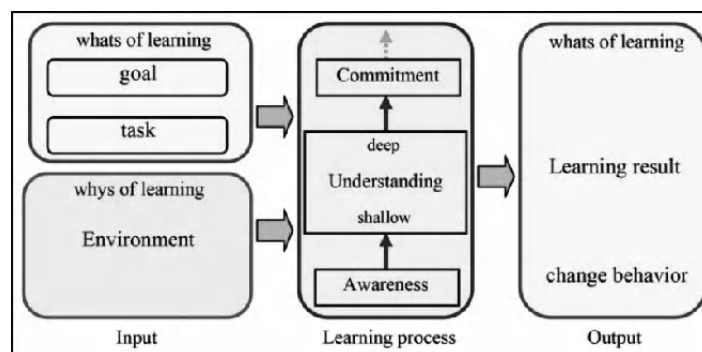


Figure 4.1. Overview of a learning process model

Stage	Meaning	The role of leadership
6 Reflection	"What have we learned?" "How have we learned?"	
5 Enactment	"I want to try this".	Allow risk taking.
4 Commitment	"I want to know about this".	Remove barriers.
3 Understanding	"I need to know about this".	Develop shared vision "whys".
2 Awareness	"I ought to know about this".	Develop shared vision "whats".
1 Ignorance	"I do not know and do not care".	Question.

Table 4.2. *The stages of learning process*

Then, we discuss requirements for modeling a learning process to achieve continuous learning. The main issue is "who generates the learning goals of a learner?" Almost all previous research in machine learning assumes that a learning goal is an external input of a learning agent or system, and it is fixed and given by the human designer [RUS 95, RUS 09]. Thus, when the learner accomplishes the learning goal, it stops learning. The same can be said for human learning. To avoid a learner stopping to learn, the leader assumes the role of leadership to guide the learning of the learner as shown in Table 4.2. Therefore, to realize the learning support system for a human to achieve continuous learning, it is necessary to automate the role of leadership, particularly to generate adequate goals as "whats" of learning.

4.1.4. How to keep supplying new goals to achieve continuous learning

As pointed out above, the main problem is how to keep supplying new goals to a learner for achieving continuous learning. A useful suggestion is from research in metalearning in education. The definition of metalearning is the state of "being aware of and taking control of one's own learning" [BIG 85]. In other words, it is the monitoring and control of learning [AND 07].

To solve this problem, we propose an idea to provide a human learner with *invisible goals* to encourage the sense of continuous awareness towards goal discovery. Figure 4.2 shows modeling a learning process by *invisible stimulus*. *Invisible stimulus* means that it has no impact on sensory perception

before action, but the response of the action differs from the past. Thus, a human learner who encounters an *invisible stimulus* enters the state of being aware of something different. There are two points to consider. The first point is that the *invisible stimulus* effects a continuous feeling in the learner's mind. The second point is that the *understanding* process of feeling something *invisible* is exposed to view. So that the output of *understanding* is a change in vision that causes a change behavior.

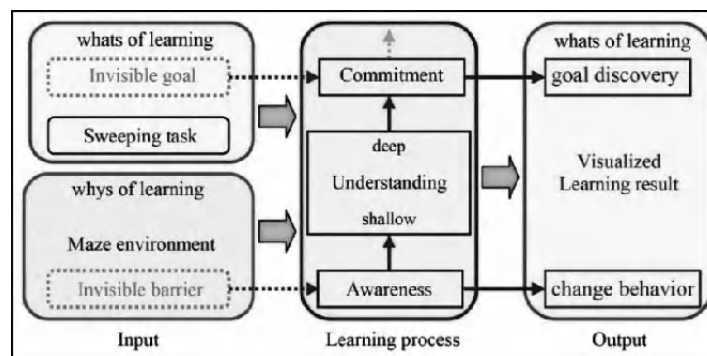


Figure 4.2. Modeling a learning process by invisible stimulus

We design two kinds of *invisible stimulus*, *invisible barrier* and *invisible goal* as shown in Figure 4.2. An *invisible barrier* makes the learner feel *awareness*, something that causes a failure of the performed action. When the learner intends to change his/her action to avoid the failure, it will be observed as a change in behavior. The *awareness* that results from the change in behavior is one of *shallow understanding*, called single-loop learning that consists of normal level learning with a fixed goal. Section 4.5 describes one of the effective usages of *invisible barriers* to guide the learner's shallow thinking or his/her trial and error to collect all solutions by a systematic search.

On the other hand, an *invisible goal* provides the learner with the unforeseen success of goal discovery. It is expected to enhance the *need* for discovering unknown goals, then it results in a goal *commitment* of the learner. *Commitment* that results in goal discovery is the one of *deep understanding*, called double-loop learning that consists of two kinds of learning level: normal level (change behavior) and metalevel (goal discovery). This chapter focuses on goal discovery for continuous learning by *invisible goals*.

4.1.5. The concept to formalize the continuous learning by a maze model

For modeling the major stages of a learning process as shown in Figure 4.2, this chapter formalizes the continuous learning by a simple maze model with *invisible goals* and designs *the maze sweeping task* that involves *multiple solutions* and *goals*. This section summarizes the design concepts of them. First, we describe the concept for designing the “whys” of learning to create a learning environment as a grid maze model, and then we describe the concept for designing the “whats” of learning as *the maze sweeping task*.

Designing a learning environment for a human learner, we adopt a grid maze model from start to goal since it is a familiar example for finding the path through a trial and error process. Figure 4.3 shows an example of a two-dimensional (2D) grid maze model. Figure 4.3(a) shows an example of the problem of a grid maze. *S* is the start state of the maze and *G* is the goal state of the maze. The details are given in the following section. For designing a learning environment, there are two points as follows:

- easy to monitor a learning process;
- capture the essential features of continuous learning.

In the first point, there are two kinds of advantages. For a learner, self-monitoring the learning processes assists in the awareness to improve them. For the system, observing the learner's learning processes enables it to evaluate the effect of the learning support system. In the second point, to evaluate a discovery learning task through the experiment with subjects within minutes, it is important to pass on easily the meaning of the experimental task to a human learner.

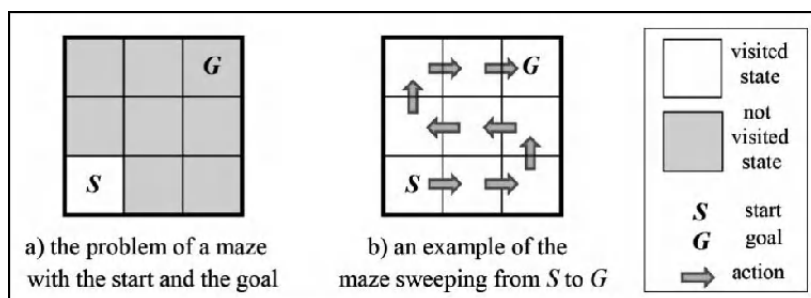


Figure 4.3. An illustrated example of a grid maze model

Next, we describe the concept for designing the “whats” of learning in a maze model. As the task: we employ *the maze sweeping* problem. *The maze sweeping* is to find the shortest path from S to G in a maze that visits all states only once. Figure 4.3(b) shows an example of the solution of the maze sweeping. For designing *the maze sweeping task* to encourage the process of continuous learning for a human learner, there are two points which are as follows:

- To drive single-loop continuous learning, *the maze sweeping task* is requested to collect all optimal solutions for each goal.
- To drive double-loop continuous learning, the goal aspects of *invisible and multiple goals* are designed.

4.2. Designing the continuous learning by a maze model

First, we describe designing the “whys” of learning to create a learning environment as a grid maze model; second, designing the “whats” of learning as the maze sweeping task is shown. After designing the thinking level, we introduce the layout design of mazes for the continuous learning task.

4.2.1. A learning environment by a maze model with invisible goals

As a learning environment, *a maze model* is defined by five elements, state set, transitions, action set, a maze task with its solution and *invisible goals*. First, we describe the structure of a 2D grid maze, and then each of the five elements is defined.

4.2.1.1. The structure of a 2D grid maze

The $n \times m$ grid maze consists of the $n \times m$ number of 1×1 squares. It is surrounded by walls. Figure 4.4 shows the structure of a 2D grid maze. Figure 4.4(a) shows a 4×3 grid maze with the start and a goal. In a grid maze, every square that touches one of their edges is connected. Figure 4.4(b) shows the X-Y coordinate of the grid maze. Each square in a maze model is called a state. Each state is distinguished by a coordinate value as (x, y) . S is the start state of a maze model that is fixed in origin $(0, 0)$. In Figure 4.4(a), G is a goal state of the maze located at $(3, 2)$.

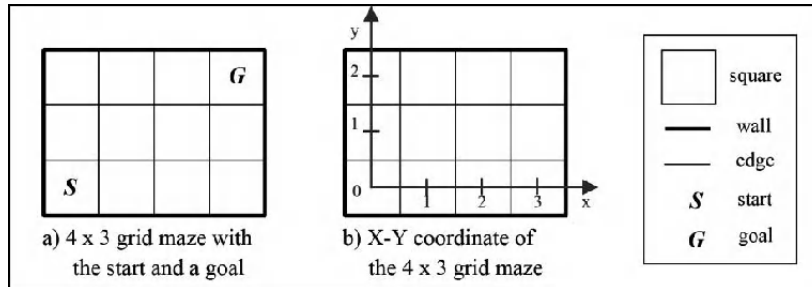


Figure 4.4. The structure of a 2D grid maze

4.2.1.2. The definition of state set

A maze model has a state set that corresponds to the structure of squares in the grid maze. Each state has several state variables as follows:

- (x, y) : X-Y coordinate value.
- Flags: they consist of S flag, G flag, invisible G flag and discovered G flag.
- The number of visits: the number of occurrences in the path from S to G .

Figure 4.5 shows an example of a maze model with invisible goals. Figure 4.5(a) shows the 3×3 maze. In each state, whether visited or not, it is distinguished by the background color of the square. A state visited just once in the path is displayed as a white square and a not visited state is displayed as a gray square.

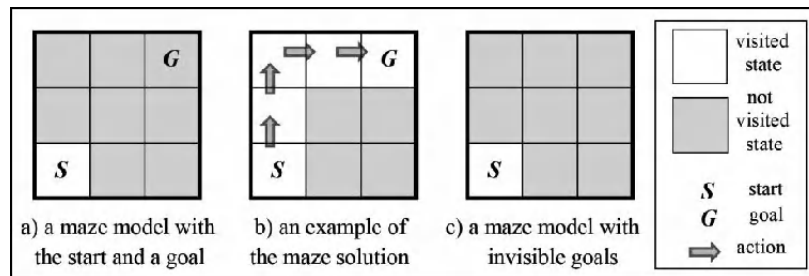


Figure 4.5. A maze model with invisible goals

4.2.1.3. *The definition of transitions and action set*

Transitions between states in a maze model is defined as whether the corresponding square are with four neighbors, {up, down, left, right} except for walls connected or not. They are represented as the labeled directed graph. The *action set* is defined as a set of labels to distinguish the possible transitions of a state. In a grid maze model with four neighbors, the *action set* is {up, down, left, right}, and each *action* of the action set is defined as follows:

- *up*: a transition to increase coordinate value of y to $y + 1$.
- *down*: a transition to decrease coordinate value of y to $y - 1$.
- *left*: a transition to decrease coordinate value of x to $x - 1$.
- *right*: a transition to increase coordinate value of x to $x + 1$.

4.2.1.4. *The definition of a maze task with its solution*

A *maze task* is defined to find the shortest path from S to G and is called a *solution* in the maze model. This task is easy since all (optimal) solutions are able to generate procedurally from relative coordinate values between S and G . A *solution* is described as the action sequence from S to G . Figure 4.5(b) shows one of the *solutions* of the maze task from S to G , and its action sequence is (up, up, right, right).

4.2.1.5. *The definition of an invisible goal*

An invisible goal is defined as the undiscovered goal state of a maze sweeping task. Figure 4.5(c) shows an example of the maze model with *invisible goals*. In this maze, the G state is hidden until a learner commits the *invisible goal* state. In this case, all states except S are possible *invisible goals*.

4.2.2. *The maze sweeping task that involves multiple goals*

4.2.2.1. *The definition of the continuous learning task*

The *continuous learning task* is defined so as to collect all *solutions* [SAT 06, YAM 11] of the task. The *optimality* of a maze task is defined as the minimum length of a path from S to G .

4.2.2.2. The definition of an achievement of the maze sweeping task

To begin with, we describe a *maze sweeping task* with a fixed goal. It is defined so as to find the (shortest) paths from S to G that visits all states only at once in the maze model. Note that S is the fixed position as shown in Figure 4.4.

An *achievement* is defined as the *continuous learning task* of a *maze sweeping* with a fixed goal. Figure 4.6 shows an example of an *achievement* of a 3×3 maze sweeping task. Figure 4.6(a) shows an initial situation of an example of an *achievement* of a 3×3 maze model. Figure 4.6(b) shows all *solutions* of the *achievement* as shown in Figure 4.6(a). An *achievement* is harder than the maze task described in section 4.2.1.4 since it needs a systematic search method to collect all *solutions*. So, it is suitable to make an acceptable difficulty of the *continuous learning task* for a human learner in a reduced size of the maze model.

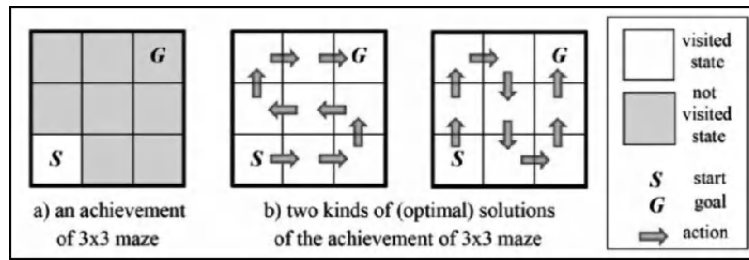


Figure 4.6. An example of an achievement of a 3×3 maze sweeping task

4.2.2.3. The definition of a stage of the maze sweeping task

In this section, we introduce a maze sweeping task that involves multiple goals. A maze sweeping task with multiple goals is defined as: to find the paths from S that visits all states only once in the maze model, note that G is the last state in the path.

A stage is defined as the continuous learning task of a maze sweeping with multiple goals. Figure 4.7 shows an example of the stage of a 3×3 maze sweeping task *under invisible goals* condition. Figure 4.7(a) shows an initial situation of the stage of a 3×3 maze *under invisible goals* condition. There are three kinds of goals in a stage of the maze model that are as follows:

- visible goal;

- invisible goal;
- dummy goal.

The first type of goal is displayed for a learner as shown in Figure 4.6(a). The second type of goal is not displayed for a learner in the beginning as shown in Figure 4.7(a). After all solutions of the corresponding invisible goal are found, it is displayed as a discovered goal (DG) as shown in Figure 4.7(b). The third type of goal is not displayed for a learner in the beginning just like an invisible goal. However, there is no solution corresponding to a dummy goal. A dummy achievement is defined as an achievement that has no solution. Each dummy goal corresponds to a dummy achievement.

As previously described in section 4.2.1.5, *invisible goals* are possible and undiscovered goal states of the *solution of a maze sweeping task*. In this stage of a 3×3 maze, there are four *invisible goal* states displayed as DG within eight states as shown in Figure 4.7(b), and other states are *dummy goal* states of this stage.

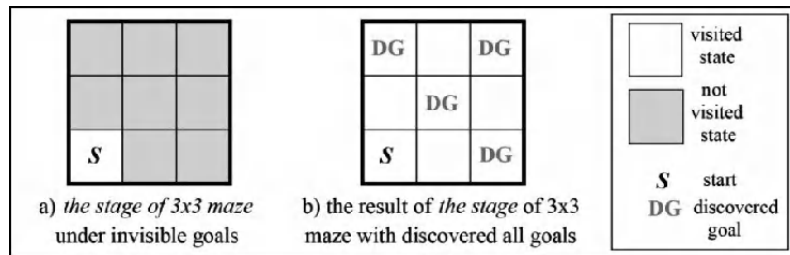


Figure 4.7. An illustrated example of the stage of a 3×3 maze sweeping task

4.2.2.4. The hierarchical structure between a stage and the achievements

Now, we describe the hierarchical structure between a *stage* and the *achievements*. Since every *stage* with more than a 2×2 maze has multiple goals, the *stage* has multiple *achievements* corresponding to the goals. Then, we show an example of the hierarchical structure between a *stage* and the *achievements*. In the case of the *stage* of a 3×3 maze under invisible goals condition as shown in Figure 4.7(a), the stage has eight states with gray except start state *S* with white. Within them, there are four *achievements* and four *dummy achievements*. Figure 4.8 shows four *achievements* of the *stage* of a 3×3 maze.

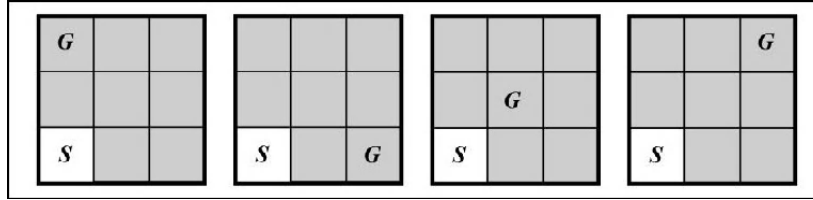


Figure 4.8. All achievements of the stage of a 3×3 maze sweeping task

4.2.3. Designing the thinking level

4.2.3.1. Objective for designing the thinking level

The objective for designing the thinking level is to keep maintaining the flow state of the human learner according to the learner's skill development. Flow is the mental state of an operation in which a person performing an activity is fully immersed in a feeling of energized focus, full involvement and enjoyment in the process of the activity [CSI 05]. We design the thinking level to reconstitute the stages of the learning process as shown in Table 4.2, in order of increasing difficulty of thinking for a human learner.

4.2.3.2. Designing the thinking level by depth and width

The thinking level is the difficulty of thinking in a learning process. Our idea is to estimate it as the number of candidates to be considered. We focus on both the constraints on learning goals and the number of solutions. Figure 4.9 shows the formalization of the thinking level space by two kinds of axis as follows:

- the width of thinking;
- the depth of thinking.

The objective of the width of thinking is to prepare alternatives for the change of the learning environment in future. It is defined by the number of solutions per achievement. We classify it whether unique or multiple. The case of multiple solutions needs a broader level of thinking to collect all of them than the case of unique solution. The objective of the width of thinking is to find goals to achieve toward continuous learning. It is defined by the condition of goals, that is whether goals are visible or *invisible*. The case of *invisible goals* is a deeper level of thinking than the case of visible goals. As

described in section 4.1.3, a learner thinks by shallow understanding (single-loop learning) under visible goals condition since the goals are given and known. On the other hand, *under invisible goals* condition, the learner thinks by deep understanding (double-loop learning) since the goals must be discovered.

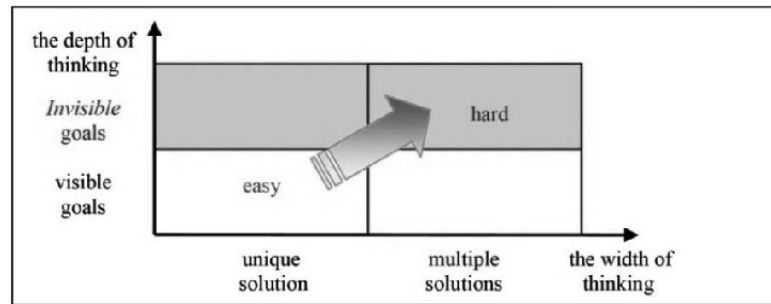


Figure 4.9. *The formalization of the thinking level space by depth and width*

4.3. The layout design of mazes for the continuous learning task

4.3.1. Overview of the continuous learning support system

Our system consists of three layers as follows:

- top level;
- maze level;
- achievement level.

Figure 4.10 shows the overview of the top level of the system for a learner. A user can opt to start or exit the experiment, select the current maze to challenge and can verify the state of progress of continuous learning by the display of several measurements described in section 4.4.1.3.

In the maze level, the user can select an achievement to challenge by clicking one of states in the current maze displayed in the center of the maze-level window. If he/she finds all the solutions in the achievement, the goal state in the achievement is displayed as DG as shown in Figure 4.7(b), and then it becomes non-selective.

Stage	1	2	3	4	total
Maze size	2 × 3	4 × 2	3 × 3	4 × 3	
Challenged Achievement	5	0	0	0	5
Score	3	0	0	0	3
Action count	29	0	0	0	29
Trial count	7	0	0	0	7
Trial time	00:00:08.441	00:00:00.000	00:00:00.000	00:00:00.000	00:00:08.441

Input Menu

Start

Current Maze Stage2

Enter Current Maze

Exit

Figure 4.10. Graphical User Interface for a learner – top level of the system

In the achievement level, the user can challenge the maze sweeping task of the achievement selected at the maze level. If the user finds a solution of the achievement, it is registered and then the system goes back to the maze level. If he/she visits *G* without finding a solution, the small window appears to indicate failure, and then he/she can restart this achievement.

4.3.2. The layout design of mazes on the thinking level space

Now, we coordinate the layout design of mazes on the thinking level space for designing the continuous learning task. Figure 4.11 shows the layout design of mazes for the continuous learning task. It is composed of four stages.

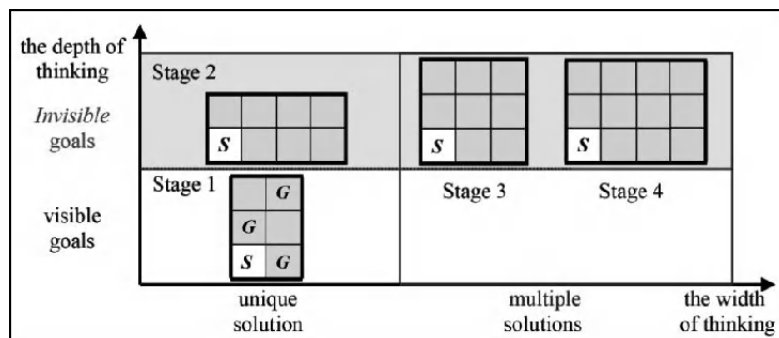


Figure 4.11. The layout design of mazes for the continuous learning task

4.3.2.1. Stage 1: visible goals under unique solution

This consists of a 2×3 maze model with three visible goals and two *dummy goals*. Each goal is linked with an achievement of the 2×3 maze with a fixed goal. The solution of each achievement with a visible goal is unique; on the other hand, each *dummy achievement* with a *dummy goal* has no solution.

4.3.2.2. Stage 2: invisible goals under unique solution

This consists of a 4×2 maze model with four *invisible goals* and three *dummy goals*. Figure 4.12 shows an overview of stage 2 under *invisible goals* condition. Figure 4.12(a) shows maze level of stage 2. Each *invisible goal* is linked with an achievement of the 4×2 maze with the corresponding *invisible goal*. The rest of each state is linked with a *dummy achievement* of the 4×2 maze with a *dummy goal*. Figure 4.12(b) shows the achievement level of stage 2. Note that for a learner, showing both goal and dummy goal is the same until the learner finds a solution on an achievement with an *invisible goal*. The solution of each achievement with an *invisible goal* is unique; on the other hand, a *dummy achievement* has no solution.

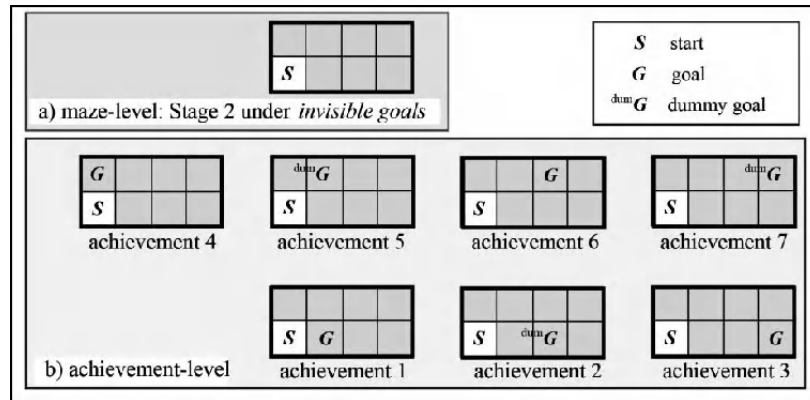


Figure 4.12. Overview of stage 2 under invisible goals condition

4.3.2.3. Stage 3: invisible goals under multiple solutions

This consists of a 3×3 maze model with four *invisible goals* as shown in Figure 4.7(b) and four *dummy goals*. As is the case with stage 2, each *invisible goal* is linked with an achievement of the 3×3 maze with the corresponding *invisible goal* as shown in Figure 4.8. Each achievement with

an invisible goal has two solutions; on the other hand, a dummy achievement has no solution. In Stage 3, there are a total of eight solutions.

4.3.2.4. Stage 4: invisible goals under many solutions

This consists of a 4×3 maze model with six *invisible goals* and five dummy goals. In this stage, there are a total of 17 solutions, much more than the number of solutions in stage 3. Within the six goals, three goals have four solutions each, two goals have two solutions each and the last goal has one solution.

4.4. Experiment

4.4.1. Experimental setup

To examine the effects of our continuous learning support system, we perform the experiment in which a total of 12 subjects are divided into two groups for comparative conditions. There are two objectives. The first one is “does our system support the continuous learning for a human?” The second question is “does the condition of invisible goals work so well to assist the continuous learning for a human?” Then, we describe the experimental task and the instruction for subjects, comparative conditions, assumptions and measurements, and the hypothesis.

4.4.1.1. The experimental task and the instruction

The experimental task explained to the subjects is to collect as many as possible of the solutions of the maze sweeping task as described in section 4.2.2. To examine the degree to work through the continuous learning for the maze sweeping task, we prepare four stages as described in section 4.3.2. All subjects are instructed as follows:

- Stage 1 is the practice maze so as to get used to the maze sweeping task.
- Stage 2 and 3 are the real part and the collection of as many solutions as possible.
- Stage 4 is a bonus maze; if you want to continue this experiment, you can challenge this stage as long as you can.

4.4.1.2. Comparative conditions

Figure 4.13 shows the experimental condition whether goals of each maze are *invisible* or not. Note that stage 1 is the common condition that all goals are visible. Figure 4.13(a) shows the condition of mazes *under invisible goals* condition. In this condition, all goals in the maze are *invisible*. Figure 4.13(b)

shows the condition of mazes under visible goals condition. In this condition, all goals in the maze are visible.

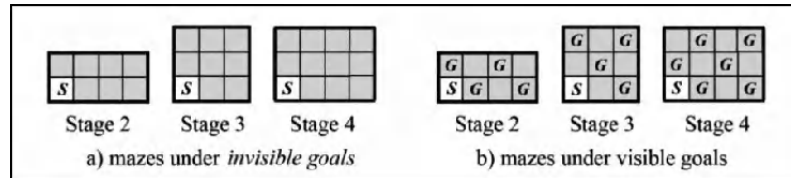


Figure 4.13. Comparison of invisible goals with visible goals

4.4.1.3. Assumptions and measurements

In this experiment, deep thinking mainly occurs at the learning stage of commitment to achieve a goal as shown in Table 4.2. It is harder to commit to a goal under invisible goals condition than to commit to a goal under visible goals condition. To measure and evaluate the continuous learning for a human, we make the following assumptions:

- The degree of continuous learning depends on the degree of depth of thinking.
- The degree of depth of thinking (double-loop learning) can be estimated by the playing time in the maze level and by the number of lines of free comments relevant to a subject's learning process in the questionnaire after the experiment.
- The degree of shallow understanding (single-loop learning) can be estimated by the playing time in the achievement level.

Next, we describe the measurements of the experiment as follows:

- 1) the number of challenged achievements;
- 2) the number of collected solutions (displayed as the score in Figure 4.10);
- 3) the number of actions of the challenged achievements;
- 4) the number of trials of the challenged achievements;
- 5) the playing time in the achievement level;
- 6) the playing time in the maze level;
- 7) the number of lines of free comments relevant to a subject's learning process in the questionnaire after the experiment.

First, four measurements are counted on each stage as shown in Figure 4.10. The fifth measurement is to estimate the degree of shallow understanding (single-loop learning). The last two measurements are to estimate the degree of deep understanding (double-loop learning) as described in section 4.3.2.

4.4.1.4. *The hypothesis*

The condition of invisible goals is expected to take deeper thinking as described in section 4.2.3, then we make a hypothesis as follows:

- The condition of invisible goals encourages deep thinking in the maze level, and it results in longer continuous learning rather than in the condition of visible goals.

4.4.2. *Experimental results*

4.4.2.1. *Does our system support the continuous learning for a human?*

This section evaluates the effectiveness of our continuous learning support system. All 12 subjects performed the bonus stage 4, and each four subjects of both conditions collected all 17 solutions of stage 4. The data of Table 4.3 and Table 4.4 is the averaged value of six subjects for each condition, and (data) is the standard deviation of six subjects for each condition. Table 4.3 shows the experimental result of the total results of stages 1, 2, 3 and 4. The seven measurements are described in section 4.4.1.3. As shown in Table 4.3 (2), about 87% of solutions (28 solutions among a total 32 solutions) are collected in both conditions.

Measurements	(1) [times]	(2) [times]	(3) [times]	(4) [times]	(5) [sec]	(6) [sec]	(7) [lines]
Conditions							
Invisible goals	17.6	28.0	341	39.8	270	1080	9.00
	(3.67)	(6.33)	(63.0)	(6.11)	(240)	(1230)	(4.00)
Visible goals	18.0	27.8	340	42.0	245	527	4.50
	(1.27)	(7.06)	(114)	(9.19)	(140)	(250)	(4.04)

Table 4.3. *The experimental result: total results of stages 1–4*

4.4.2.2. Does the condition of invisible goals work so well to assist the continuous learning for a human?

This section evaluates the effectiveness of the condition of *invisible goals* compared to the condition of visible goals. In Table 4.3, there is no significant difference in the first five measurements (1), (2), (3), (4) and (5) between both conditions. However, the last two measurements (6) and (7) that are relevant to the degree of depth of thinking (double-loop learning) seem to be different. Analyzing the ratio of (6) divided by (5), four of six subjects are over four times under *invisible goals* condition, compared to one of six subjects over four times under visible goals condition.

Next, we analyze the measurements in stage 4 to evaluate the degree of continuous learning in a straightforward way. Table 4.4 shows the experimental result of stage 4. The six measurements are the same as in Table 4.3. In Table 4.4, there is no significant difference in first four measurements (1), (2), (3) and (4) between both conditions. However, the last two measurements (5) and (6), both of the results under *invisible goals* condition are longer than the results under visible goals condition. Therefore, these results suggest that the *invisible goals* condition is more effective to assist the deep understanding in a learning process than the visible goals condition.

Measurements	(1) [times]	(2) [times]	(3) [times]	(4) [times]	(5) [sec]	(6) [sec]
Conditions						
Invisible goals	6.50	14.0	214	21.3	201	648
	(2.95)	(5.62)	(67.9)	(6.74)	(243)	(1045)
Visible goals	6.00	14.0	219	21.7	112	199
	(0.63)	(5.29)	(112)	(10.4)	(65.0)	(172)

Table 4.4. The experimental result of stage 4

4.5. Discussions

4.5.1. The role of motivations to drive the continuous learning

To begin with, we discuss the role of motivations to drive the continuous learning. In the experiment, all 12 subjects performed the bonus stage 4, and 8 of 12 subjects collected all 17 solutions of stage 4. The reason may be the

display effect of DG in the maze level when all solutions corresponding to G state are collected as described in section 4.3.1. Figure 4.14 shows the hierarchical model of achievement motivation [ELL 97, ELL 06]. According to this model, the display of DG positively effects competence expectancy since he/she accomplished the achievement. As a result, it increases the intrinsic motivation to the task.

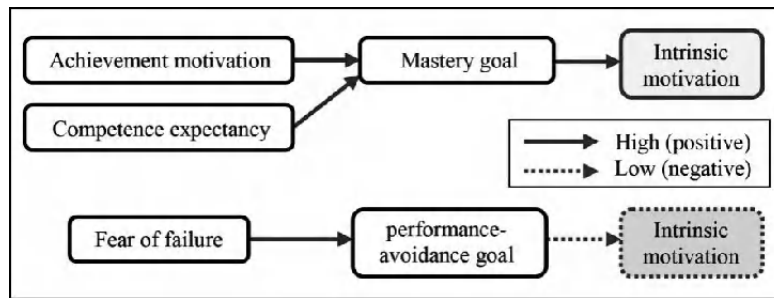


Figure 4.14. Hierarchical model of achievement motivation

4.5.2. Why is it important to collect all solutions for continuous learning?

We discuss the reason why to collect all solutions for continuous learning. A useful idea is from the economic complexity index [HID 09, HAU 11]. They present the measures of complexity of countries on economic growth as the productivity of a country resides in the diversity of its available non-tradable capabilities in that country. They also show that the measures of complexity are correlated with a country's level of income (GDP), and that deviations from this relationship are predictive of future growth.

What is interesting about this is that they take it as an analogy of "Lego[®] theory" [HID 09]. In that, economic growth is like assembling many hundreds of different Lego[®] pieces. In other words, a child who has various lego pieces in his/her bucket is able to make more complex Lego[®] model. We can say the same for the continuous learning. A learner who has various solutions in his/her learning experience is able to make more complex solution according to the growth of his/her continuous learning process.

Figure 4.15 shows the example of reusing previous solutions to solve a 4×3 maze sweeping task. Figure 4.15(a) shows the 4×3 maze *under invisible goals* condition as described in section 4.3.2. The solution of this is

classified into five cases according to the beginning of the action sequence as shown in Figures 4.15(b)–(f) as follows:

(b) (up, up, right): there are four DGs and the number of solutions is eight by reusing the learning result (solutions) of a 3×3 maze.

(c) (up, right): there is no solution.

(d) (right, up): there are four DGs and the number of solutions is four.

(e) (right, right, up): there is one DG and one solution.

(f) (right, right, right, up): there are four DGs and the number of solutions is four by reusing the learning result (solutions) of the 4×2 maze.

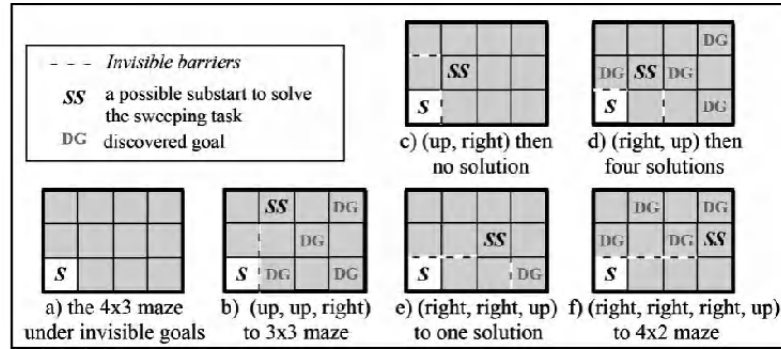


Figure 4.15. Solving the 4×3 maze by reusing previous solutions

Next, we show an idea of the effective usage of *invisible barriers* to guide the learner's shallow thinking or his/her trial and error to collect all solutions by systematic search. In Figure 4.14, the dotted line is the *invisible barriers* in order to guide the beginning of the action sequence of a learner. It is expected to result in the awareness for shallow thinking or learning through trial and error. As part of our future work, we are planning to examine this effective usage of *invisible barriers*.

4.5.3. Toward an application of a maze model and invisible goals

This section discusses the meaning of *invisible goals* toward the application of a maze model. Regarding the X-Y coordinate of the maze model as shown in Figure 4.4(b), we look at the x-axis as the previous values

and the y -axis as the new values, and then we discuss the *invisible goals* on the development of an adhesive as typical discovery episode.

First, we begin with the development history of the Post-it note [ROB 89]. In 1968, a researcher of 3M was attempting to develop a superstrong adhesive, but instead he accidentally created a “low-tack”, reusable, pressure-sensitive adhesive. Figure 4.16 shows the simple functional map of adhesive in 1968. In this single axis of values, a state without “(a) development goal” is regarded as a failure as shown in “(b) product failure”. However, he focused on the reusable function of the weak adhesive. Figure 4.17 shows the extended functional map of the adhesive, in which the x -axis shows the previous values as the function of the adhesion force and y -axis shows a new value as the function of reusability.

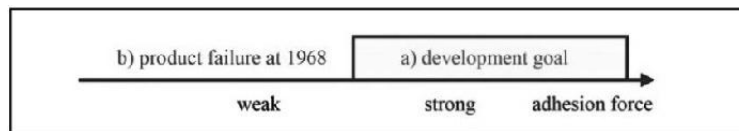


Figure 4.16. *The simple functional map on adhesive*

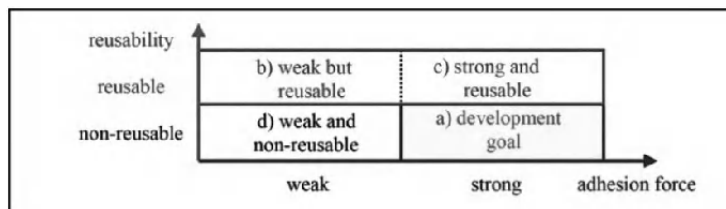


Figure 4.17. *Extended functional map on adhesive*

Second, we discuss the transition from focusing on a new function to discovering an unknown usage. The key point is “how to discover the potential needs behind the new function?” In 1974, a colleague of the researcher who created a “low-tack”, reusable adhesive came up with the idea of using the adhesive to anchor his bookmark in his hymnbook. At this time, about “(b) weak but reusable” as shown in Figure 4.17, a new usage for temporarily attaching is discovered. In 1978, 3M issued free samples designed for temporarily attaching notes to documents and other surfaces and in 1980, the product debuted in US stores as “Post-it notes”. Now, this episode

suggests that an *invisible goal* in this case is revealed by the discovery of a different usage from previously known goals.

Then, look at the rest states of (c) and (d) in Figure 4.17, what kind of invisible goals are hidden? Table 4.5 summarizes the R&D process of the adhesive materials from extended function map to discovering an unknown usage. In Table 4.5, (c) is a adhesive material with strong and reusable functions. In 2012, a new adhesive material with these functions is presented [AKI 12]. The idea is that the new material becomes liquid under ultraviolet light and it becomes solid under visible light. The usage of (c) is not found yet, it is an interesting continuous discovery with Post-it notes.

Function map	A new material with new functions	The discovery of a new usage
(b) Weak but reusable	Accidentally created in 1968	1974: Temporally attachment 1980: Post-it notes
(c) Strong and reusable	R&D in 2012	
(d) Weak and non-reusable	(Difficult to determine bonding strength objectively)	

Table 4.5. *The R&D process from extended function map to discovering an unknown usage*

4.6. Conclusions

We described the learning support system based on a learning process model to guide a human to achieve continuous learning. To support the awareness and understanding of learning processes, we proposed to provide a human learner with *invisible goals* to encourage the sense of continuous awareness toward goal discovery. Then, we formalized the continuous learning by a simple maze model with *invisible goals* and designed *the maze sweeping task* that involves *multiple solutions* and *goals*. The experimental results showed that our continuous learning support system is effective. Moreover, besides *the invisible goals* is more effective to assist deep understanding in a learning process than the visible goals.

As part of our future work, we are planning to quantitate the degree of difficulty of continuous learning as the complexity of maze model and action sequences of a learner for maintaining the flow state of the human learner according to the learner's skill up.

4.7. Acknowledgments

The authors would like to thank Professors Habib and Shimohara for offering a good opportunity to present this research. This work was supported by JSPS KAKENHI (Grant-in-Aid for Scientific Research (C)) grant number 23500197.

4.8. Bibliography

- [AKI 12] AKIYAMA H., YOSHIDA M., "Photochemically reversible liquefaction and solidification of single compounds based on a sugar alcohol scaffold with multi azo-arms", *Advanced Materials*, vol. 24, no. 17, pp. 2353–2356, 2012.
- [AND 07] ANDERSON M.L., OATES T., "A review of recent research in metareasoning and metalearning", *AI Magazine*, vol. 28, no. 1, pp. 7–16, 2007.
- [BIG 85] BIGGS J.B., "The role of meta-learning in study process", *British Journal of Educational Psychology*, vol. 55, no. 3, pp. 185–212, 1985.
- [BUC 96] BUCKLER B., "A learning process model to achieve continuous improvement", *The Learning Organization*, vol. 3, no. 3, pp. 31–39, 1996.
- [CSI 05] CSIKSZENTMIHALYI M., ABUHAMDEH S., NAKAMURA J., "Flow", in ELLIOT A.J., DWECK C.S. (eds), *Handbook of Competence and Motivation*, The Guilford Press, New York, pp. 598–698, 2005.
- [ELL 97] ELLIOT A.J., CHURCH M.A., "A hierarchical model of approach and avoidance achievement motivation", *Journal of Personality and Social Psychology*, vol. 72, no. 1, pp. 218–232, 1997.
- [ELL 06] ELLIOT A.J., "The hierarchical model of approach-avoidance motivation", *Motivation and Emotion*, vol. 30, no. 2, pp. 111–116, July 2006.
- [HAB 06] HABIB M.K., "Mechatronics engineering the evolution, the needs and the challenges", *Proceedings of the 32nd Annual Conference of IEEE Industrial Electronics Society (IECON 2006)*, Institute of Electrical and Electronics Engineers (IEEE), pp. 4510–4515, 2006.
- [HAB 08] HABIB M.K., "Interdisciplinary mechatronics: problem solving, creative thinking and concurrent design synergy", *International Journal of Mechatronics and Manufacturing Systems*, vol. 1, no. 1, pp. 264–269, 2008.

- [HAR 99] HARDLESS C., NULDEN U., “Visualizing learning activities to support tutors”, *Proceedings of the CHI '99 Extended Abstracts on Human Factors in Computing Systems*, ACM, New York, pp. 312–313, 1999.
- [HAU 11] HAUSMANN R., HIDALGO C.A., *et al.*, *The Atlas of Economic Complexity*, Puritan Press, 2011.
- [HID 09] HIDALGO C.A., HAUSMANN R., “The building blocks of economic complexity”, *Proceeding of the National Academy of Sciences (PNAS)*, vol. 106, no. 26, pp. 10570–10575, June 2009.
- [MAY 11] MAY M., GEORGE S., PRÉVÔT P., “TrAVis to enhance students’ self-monitoring in online learning supported by computer-mediated communication tools”, *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 3, pp. 623–634, 2011.
- [PIN 09] PINK D.H., *Drive: The Surprising Truth About What Motivates Us*, Riverhead Books, 2009.
- [RAU 96] RAUTERBERG M., AEPPLI R., “How to measure the behavioural and cognitive complexity of learning processes in man-machine systems”, *Educational Multimedia and Hypermedia--ED-MEDIA'96*, Charlottesville, VA, pp. 581–586, 1996.
- [ROB 89] ROBERTS R.M., *Serendipity: Accidental Discoveries in Science*, Wiley, 1989.
- [RUS 95] RUSSELL S., NORVIG P., *Artificial Intelligence: A Modern Approach*, 1st ed., Prentice Hall, January 1995.
- [RUS 09] RUSSELL S., NORVIG P., *Artificial Intelligence: A Modern Approach*, 3rd ed., Prentice Hall, December 2009.
- [SAT 06] SATOH K., YAMAGUCHI T., “Preparing various policies for interactive reinforcement learning”, *Proceedings of the SICE-ICASE International Joint Conference 2006 (SICE-ICCAS 2006)*, Busan, Korea, pp. 2440–2444, October 2006.
- [SKL 10] SKLAR E., RICHARDS D., “Agent-based systems for human learners”, *The Knowledge Engineering Review*, vol. 25, no. 2, pp. 111–135, June 2010.
- [SUZ 10] SUZUKI S., “Human adaptive mecatronics”, *IEEE Industrial Electronics Magazine*, vol. 4, no. 2, pp. 28–35, 2010.
- [YAM 11] YAMAGUCHI T., NISHIMURA T., SATO K., “How to recommend preferable solutions of a user in interactive reinforcement learning?”, in MELLOUK A. (ed.), *Advances in Reinforcement Learning*, InTech Open Access Publisher, pp. 137–156, 2011.

Chapter 5

PWM Waveform Generation Using Pulse-Type Hardware Neural Networks

This chapter presents the pulse-type hardware neural networks (P-HNN) which could output the pulse width modulation (PWM) waveform of the PWM servo motor for robots. The basic component of the P-HNN is the pulse-type hardware neuron model (P-HNM). The P-HNM generates oscillatory patterns of electrical activity similar to those of living organisms. Basic components of the P-HNM consist of the cell body model, the axon model and the synaptic model. The P-HNM has the same basic features of biological neurons, such as threshold, refractory period and spatiotemporal summation characteristics, and enables the generation of continuous action potentials. In this chapter, we used the synaptic model which could vary the synaptic weights by controlling voltage. The P-HNN could output several pulse widths of the PWM waveform by varying the synaptic weights. As a result, we show that P-HNN could control the PWM servo motor from 0° to 180° . The P-HNN realized the control of the PWM servo motor without using any software programs, or A/D converters. In addition, it is shown that the pulse waveform generation method of the P-HNM is the same as for living organisms.

5.1. Introduction

The PWM controlling system was often used for several types of robot controlling systems. The PWM controlling system was easily controlled by the programmed control. Therefore, the programmed control by microcontroller has been the dominant system among robot controlling systems, such as locomotion controlling system or sensory information processing system [KAZ 98]. In the future, a robot controlling system will require an increasingly higher functionalization to play an important role in human living space. However, higher functionalization of a programmed control is difficult because the microcontroller has a limitation in memory capacity. Therefore, operational control by an operator was undertaken in complicated or unprogrammed situations. These robot controlling systems were not flexible autonomous controlling systems. On the other hand, neural networks (NN) of living organisms have excellent information processing functions for controlling systems [KIY 87, GEN 95, SIM 00, NAO 01, GIS 01]. This is because living organisms make excellent decisions to control locomotion according to the sensory information from external situations. For this reason, various studies are investigating the implementation of neural computers using new algorithms based on the principles of living brains. In particular, oscillatory patterns of electrical activity are a ubiquitous feature in nervous systems. Living organisms use several oscillatory patterns for movement, swallowing, heart rhythms, etc. [DEL 80, MIC 02]. Coupled neural oscillators are drawing attention to clarify oscillatory patterns. Synchronization phenomena or bifurcation phenomena of coupled neural systems have been studied using the Hodgkin–Huxley model or the Bonhoeffer–van der Pol model, which is a mathematical class II neuron [KUN 03, KUN 06, SHI 07]. In addition, Fujitsu Automation developed the small humanoid robot: HOAP-1, HOAP-2 and HOAP-3, which can generate robotic motion using the mathematical neural network (NeuROMA) [JIA 02, RIA 02]. NeuROMA is based on the biological central pattern generator (CPG), which is a locomotion controller for living organisms.

Earlier, various supercomputers were constructed for developing the information processing capability to calculate the research which has large computational effort (<http://www.top500.org/>). Nevertheless, to simulate the excellent information processing functions of living organisms, the computational effort becomes great using a high-nonlinearity neuron model. Therefore, the mathematical neuron model in large-scale NN is difficult to process in continuous time because the simulation is limited by

the computer's performance, such as the processing speed and the memory capacity. In contrast, hardware models are advantageous because, even if a circuit scale becomes large, a nonlinear operation can be processed at high speed. Therefore, the construction of a hardware model that can generate oscillatory patterns is desired because nonlinear operations can be processed at higher speeds than the mathematical model. The hardware ring coupled oscillators have already been studied as a system which can demonstrate the various oscillatory patterns and the synchronization phenomena [END 78, KIT 01]. For this reason, the ring coupled oscillators are expected to be a structural element of the cellular NN. However, most of the hardware models contain the inductor in circuit architecture [END 78, KIT 01, YAM 99, YAM 03]. If the system contains the inductor on the circuit system, it is difficult to implement the system to a complementary metal-oxide semiconductor integrated circuit (CMOS IC) chip.

We are researching the implementation of the hardware NN for a robot controlling system using HNMs [MAT 02, SAE 03, SAS 05, NAK 04, HAT 06, KEN 12a, KEN 12b]. The P-HNM has the same basic features of biological neurons, such as threshold, refractory period and spatiotemporal summation characteristics, and enables the generation of continuous action potentials. Furthermore, the P-HNM can oscillate without an inductor, therefore it is easy to implement the system to a CMOS IC chip [MAT 02, SAE 03]. Previously, we proposed the coupled neural oscillators using the P-HNMs. It was constructed by cell body models and synaptic models which have excitatory and inhibitory synaptic connections. We described the oscillating phenomena in an autonomous system [NAK 04, HAT 06]. This chapter presents the P-HNN which could output the PWM waveform of the PWM servo motor for robots.

In this chapter, first, we will show the examples of pulse specifications of the PWM servo motor. Second, we discuss the circuit diagram of the P-HNM and its basic characteristics. Third, we construct a P-HNN, and finally we will show results of the PWM servo motor control by the P-HNN.

5.2. PWM servo motor

Many robots used the PWM servo motor as an actuator to locomote the robot system. The PWM servo motor could be easily controlled by the programmed control by microcontroller. However, in the case of a more advanced robot controlling system compared with microcontroller controlling system, the robot system carries a computer controlling system or is wired by

cable to the computer controlling system [KAZ 98]. Consequently, advanced robot control by a compact system is desired. This section describes the examples of the PWM servo motor.

Figure 5.1 shows the example of pulse waveform to actuate the PWM servo motor. It is necessary to output the pulse waveform to control the servo motor. For example, we used the HRS-8498HB by Hitec Multiplex Japan for the PWM servo motor. Pulse specifications of a servo motor are shown in Table 5.1. These specifications are the measured control signal using an oscilloscope. It is shown that if the P-HNN can output the waveform such as shown in Figure 5.1 and satisfy the specifications of Table 5.1, the PWM servo motor could be controlled.

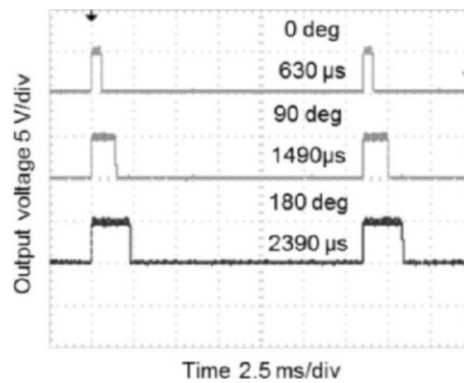


Figure 5.1. Pulse waveform to actuate the servo motor

Hitec Multiplex Japan HSR-8498HB	
Pulse period	16,040 μs
Peak-to-peak voltage	5.04 V
Motion range	0–180°
Increase of pulse width per degree	10 μs
Minimum pulse width (0°)	600 μs
Maximum pulse width (180°)	2,400 μs

Table 5.1. Pulse specifications of servo motor

5.3. Pulse-type hardware neuron model

The P-HNMs are used for the basic elements of the P-HNN. The P-HNM consists of a synaptic model, cell body model and axon model. In this section, we will show the circuit diagrams and consider the basic characteristics of the P-HNMs. The simulation results were given by OrCAD.

5.3.1. Basic cell body model

Figure 5.2 shows the circuit diagram of the basic cell body model. The cell body model consists of a voltage control type negative resistance circuit, an equivalent inductance circuit, and a membrane capacitor C_M and leak resistor R_M . The voltage control type negative resistance circuit with equivalent inductance circuit consists of n-channel enhancement metal-oxide semiconductor field-effect transistor (MOSFET), p-channel enhancement MOSFET, voltage source V_A , resistor R_G and capacitor C_G . The cell body model has the negative resistance property that changes with time like a biological neuron, and enables the generation of a continuous action potential $v_M(t)$ by a self-excited oscillation and a separately excited oscillation. Moreover, the cell body model can switch between both oscillations by changing V_A . The separately excited oscillation occurs due to pulse train stimulus. In other word, the cell body model is an oscillator that has a voltage control type negative resistance circuit.

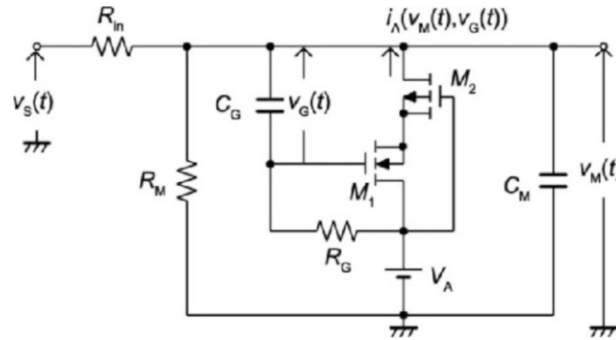


Figure 5.2. Circuit diagram of basic cell body model

The cell body model outputs the output voltage $v_M(t)$ in the case of inputting the input voltage $v_s(t)$, where $v_M(t)$ is the voltage at the end of capacitor C_M . $i_A(v_M(t), v_G(t))$ shows the nonlinear current, which flows in the

negative resistance circuit for output voltage, where $v_G(t)$ is the voltage at the end of capacitor C_G . Output voltage $v_M(t)$ and $v_G(t)$ were oscillatory waveforms which can be described by following simultaneous differential equations:

$$C_M \frac{dv_M(t)}{dt} = \frac{v_S(t)}{R_{in}} - \frac{v_M(t)}{R_M} - \frac{v_M(t) - v_G(t) - V_A}{R_G} + i_A(v_M(t), v_G(t)) \quad [5.1]$$

$$C_G \frac{dv_G(t)}{dt} = \frac{v_M(t) - v_G(t) - V_A}{R_G} \quad [5.2]$$

Nonlinear function $i_A(v_M(t), v_G(t))$ is shown as follows:

In the case of $v_G(t) + V_A + V_{Tn} + V_{Tp} < v_M(t) \leq V_A + V_{Tp}$,

$$i_A(v_M(t), v_G(t)) = \frac{\beta}{8} (A + B)^2 \quad [5.3]$$

In the case of $V_A + V_{Tp} < v_M(t) \leq v_G(t) + V_A + V_{Tn}$,

$$i_A(v_M(t), v_G(t)) = \frac{\beta (A)^2 (A - 2B)^2}{8 (A + B)^2} \quad [5.4]$$

In the case of $v_G(t) + V_A + V_{Tn} < v_M(t) \leq V_A$,

$$i_A(v_M(t), v_G(t)) = \left\{ (V_A - v_M(t))(V_A - v_M(t) + 2A)(V_A + v_M(t) - 2V_{Tp}) \right. \\ \left. (V_A - 3v_M(t) + 2V_{Tp} + 2A) \right\} / 8 (A + B)^2 \quad [5.5]$$

where $A = v_G(t) + V_{Tn}$, $B = V_{Tp} - v_M(t)$, V_{Tn} and V_{Tp} are threshold voltages of n- and p-channel MOSFETs, and k_n and k_p are conductance constants of n- and p-channel MOSFETs, respectively. In addition, we approximate the k_n and k_p as follows:

$$k_n = k_p = \frac{\mu \epsilon}{2t} \frac{W}{L} = \frac{\beta}{2} \quad [5.6]$$

where μ is the carrier mobility, ε is the dielectric constant of gate oxide, t is the oxide film thickness, W is the channel width and L is the channel length.

Figure 5.3 shows an example of the phase plane of the circuit in Figure 5.2. The circuit parameters are $\beta = 0.23 \text{ mA/V}^2$, $V_{Tn} = 0.8 \text{ V}$, $V_{Tp} = 1.5 \text{ V}$, $C_M = 10 \text{ pF}$, $C_G = 20 \text{ pF}$, $R_M = 10 \text{ k}\Omega$, $R_G = 300 \text{ k}\Omega$ and $V_A = 3.3 \text{ V}$. The abscissa is $v_M(t)$ and the ordinate is $v_G(t)$. The dotted line is $v_M(t)$ -nullcline, the broken line is $v_G(t)$ -nullcline and the solid line is the attractor, as shown in Figure 5.3. The attractor is drawn in a limit cycle. Moreover, Figure 5.3 shows that the cell body model has a characteristic of the class II neuron such as the Hodgkin–Huxley model or the Bonhoeffer–van der Pol model. In this chapter, we use the cell body model as a basic element of the coupled neural oscillator.

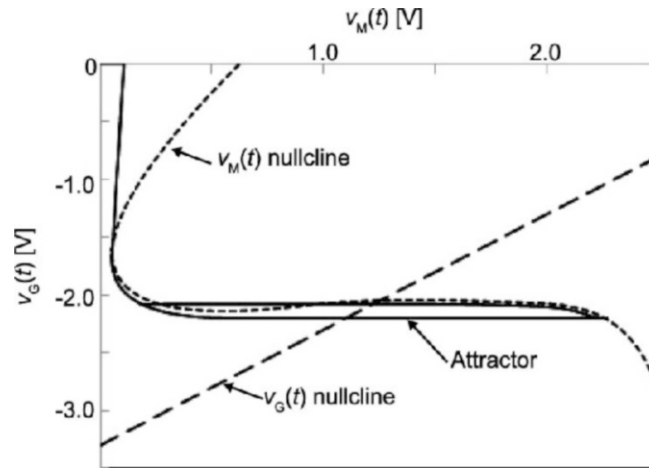


Figure 5.3. Phase plane of basic cell body model

5.3.2. Cell body model for PWM controlling system

The cell body model shown in Figure 5.2 is for CMOS IC circuit. The cell body model shown in Figure 5.4 is for a discrete circuit. Hereafter, the basic cell body model indicates the Figure 5.2 circuit and the cell body model indicates the Figure 5.4 circuit. In these circuits, we use BSS83 ([©]Koninklijke Philips Electronics N.V.) as an n-channel enhancement MOSFET M_{C1} and BSH205 ([©]Koninklijke Philips Electronics N.V.) as a

p-channel enhancement MOSFET M_{C2} . We add the bias circuit to the gate terminal of M_{C2} . The basic characteristics of Figure 5.4 and Figure 5.2 are equivalent. The circuit parameters are $C_{CM} = 2.2$ nF, $C_{CG} = 1.2$ nF, $R_{CM} = 10$ k Ω , $R_{CG} = 390$ k Ω , $R_{Ca} = 20$ k Ω , $R_{Cb} = 15$ k Ω and $V_A = 3.3$ V.

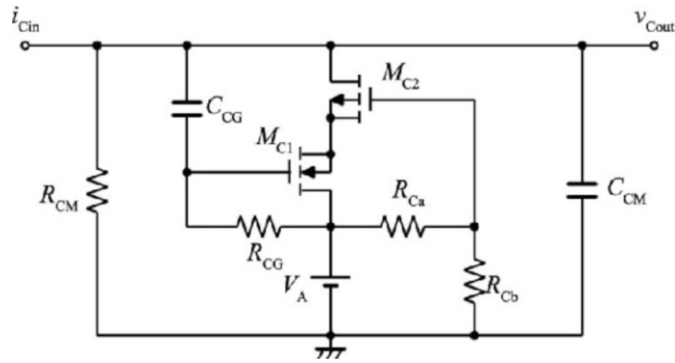


Figure 5.4. Circuit diagram of cell body model for PWM controlling system

Figure 5.5 shows the oscillatory characteristic of the Figure 5.4 circuit. This figure shows a cell body model that could oscillate in $3.4 \leq V_A \leq 4.7$ by self-excited oscillation. In this chapter, we set the cell body model to separately excited oscillation mode. Therefore, we set as $V_A = 3.3$ V.

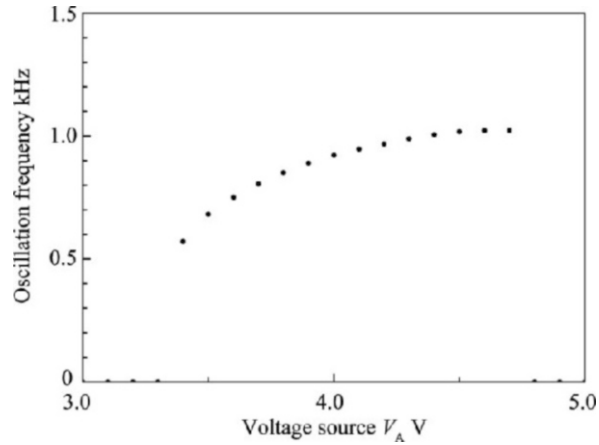


Figure 5.5. Oscillatory characteristic of cell body model

5.3.3. Synaptic model

Figure 5.6 shows the circuit diagram of a synaptic model. The synaptic model has the spatiotemporal summation characteristics similar to those of living organisms. The synaptic model spatiotemporally summed the outputs of cell body models according to synaptic weights. Synaptic weights are controlled by the synaptic weight control voltage V_{SC} . The circuit parameters of synaptic model are as follows: $C_s = 1$ pF; $M_{S1}:W/L = 3$; $M_{S2}, M_{S3}, M_{S7}, M_{S8}:W/L = 1$; $M_{S4}:W/L = 0.25$; $M_{S5}:W/L = 5$; and $M_{S6}:W/L = 0.2$. The voltage source is $V_{DD} = 5$ V.

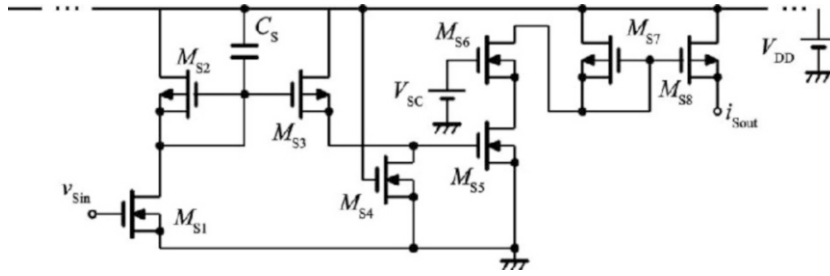


Figure 5.6. Circuit diagram of synaptic model

Figure 5.7 shows an example of output current i_{Sout} of the synaptic model in the case of a varying synaptic weight control voltage V_{SC} . This figure shows that the i_{Sout} could change by varying V_{SC} .

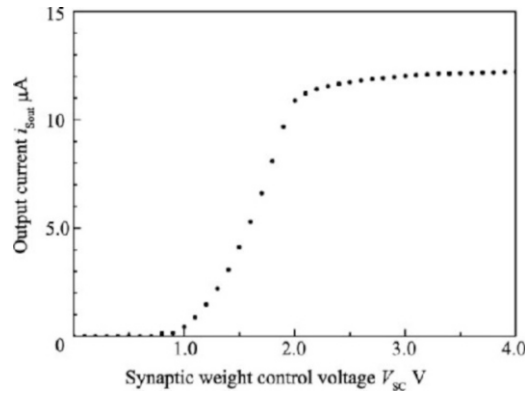


Figure 5.7. Output current characteristic of synaptic model varying synaptic weight control voltage

5.3.4. Axon model

Figure 5.8 shows the circuit diagram of an axon model. The axon models are a connected cascade. The axon model is an actively by distributed constant line which has the threshold function and waveform shaping function, just like living organisms. The input current of the axon model is i_{Ain} . The circuit parameters of the axon model are as follows: $C_{Mm} = 3.9$ nF, $C_{Gm} = 1.5$ nF, $R_{Mm} = 10$ k Ω , $R_{Gm} = 390$ k Ω , $R_{am} = 20$ k Ω and $R_{bm} = 15$ k Ω . The voltage sources are $V_A = 3.3$ V and $V_{DD} = 5$ V.

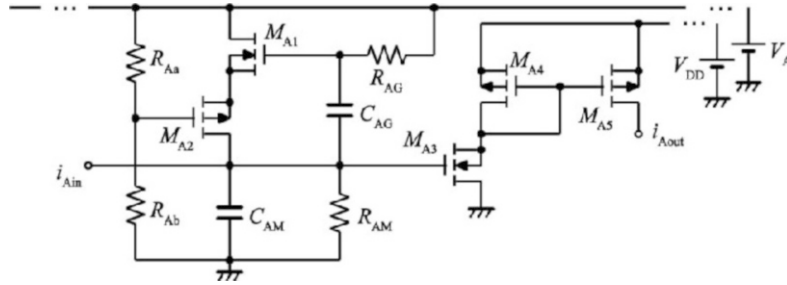


Figure 5.8. Circuit diagram of axon model

5.4. Pulse-type hardware neural networks

Figure 5.9 shows the schematic diagram of P-HNN. The P-HNN consists of cell body models, synaptic models and axon models. In the figure, C indicates a cell body model, A indicates an axon model and S indicates a synaptic model. The P-HNN consists of 24 cell body models, 25 synaptic models and 40 axon models. In the case of inputting the external trigger pulse to C_1 , the pulse propagates the ring NN with delay time. One cell body model generates a pulse width of approximately 200 μ s. Delay time of pulses is controlled by the synaptic weights of S_n . By varying the synaptic weights, the PWM waveform width could change from 600 μ s to 2,400 μ s. However, the total length of the P-HNN will change with varying synaptic weights. Therefore, C_1' and S_1' will adjust the total length of the output waveform. In the case of increasing the synaptic weights of S_n , the synaptic weights of S_n' will decrease.

In the following, we will consider the basic characteristics of the P-HNN.

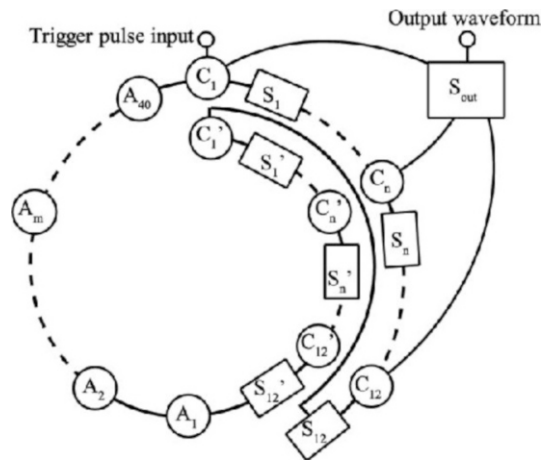


Figure 5.9. Schematic diagram of the P-HNN

Figure 5.10 shows the delay mechanism by varying the synaptic weights. In the case where synaptic weights are small, the input current of a cell body model is also small. Therefore, the pulse rise time will be longer. On the other hand, in the case where synaptic weights are large, the input current of a cell body model is also large. Thus, the pulse rise time will be shorter.

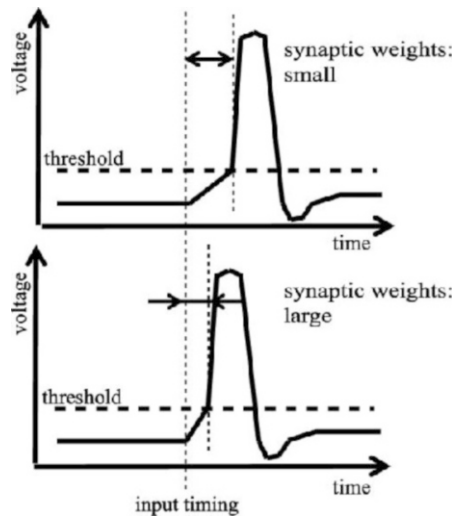


Figure 5.10. Delay mechanism by varying the synaptic weights

Figure 5.11 shows the delay time characteristic with varying synaptic weight control voltage V_{SC} . In the figure, the cell body model 1 (C_1) is connected to the cell body model 2 (C_2) by synaptic model (S_1). The delay time is the pulse interval between the output pulse of C_1 and C_2 . It is shown that the delay time could change by varying the synaptic weights.

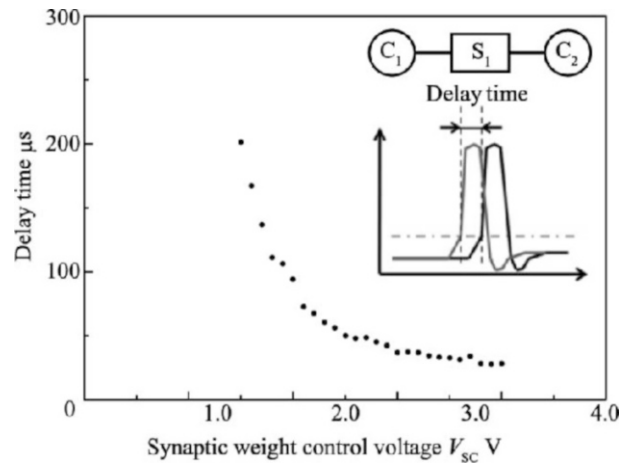


Figure 5.11. Delay time characteristic with varying synaptic weight control voltage

Figure 5.12 shows the mechanism for changing the pulse width of the PWM waveform. The pulses that have 200 μ s pulse width are outputted by C_n ($1 \leq n \leq 12$). Therefore, changing the delay time by varying the synaptic weight could change the total length of 12 pulses. The synaptic model S_{out} of Figure 5.9 has added 12 pulses.

Figure 5.13 shows the pulse width of the PWM waveform and the angle of the PWM servo motor in the case of varying synaptic weight control voltage V_{SC} . In this figure, open circles indicate the pulse width and solid circles indicate the angle of the PWM servo motor. This figure shows that the P-HNN can change the pulse width of the output waveform by changing the synaptic weight control voltage.

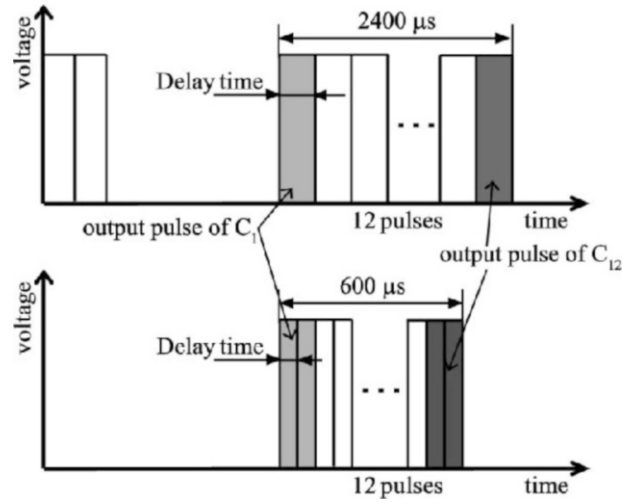


Figure 5.12. Mechanism of changing the pulse width of the PWM waveform

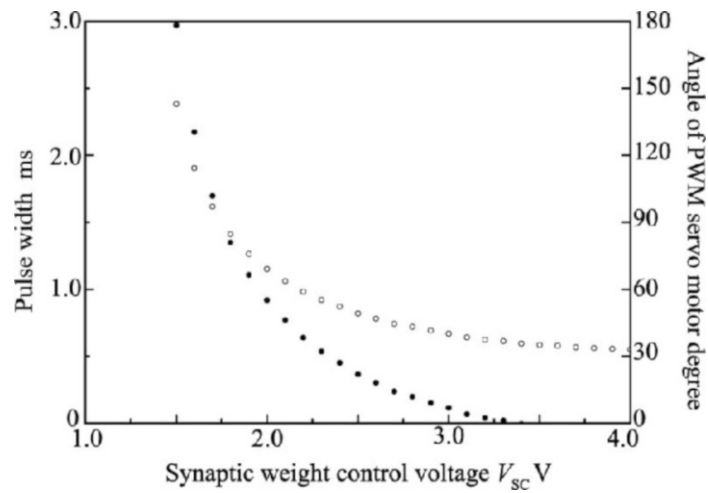


Figure 5.13. Pulse width of the PWM waveform and angle of the PWM servo motor in the case of varying synaptic weight control voltage V_{sc}

Figure 5.14 shows an example of the PWM waveform. It has been shown that the constructed P-HNN could output the PWM waveform by varying synaptic weight control voltage.

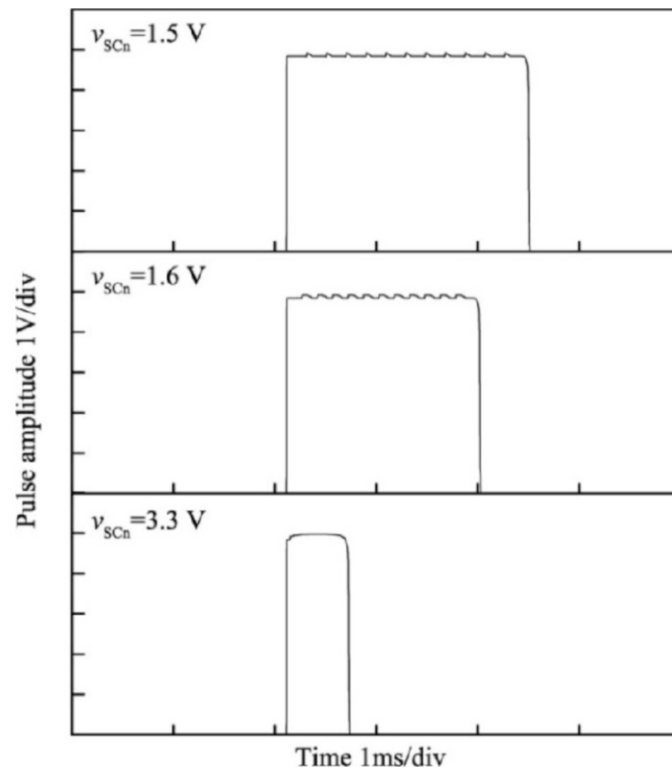


Figure 5.14. Example of the generated PWM waveform

5.5. Measurements of constructed discrete circuit

The measured results of the cell body model are shown in Figure 5.15. It is constructed by a discrete circuit. This figure shows that the pulse width of the constructed cell body model is longer than the designed value. Therefore, we have to adjust the parameters of the cell body model.

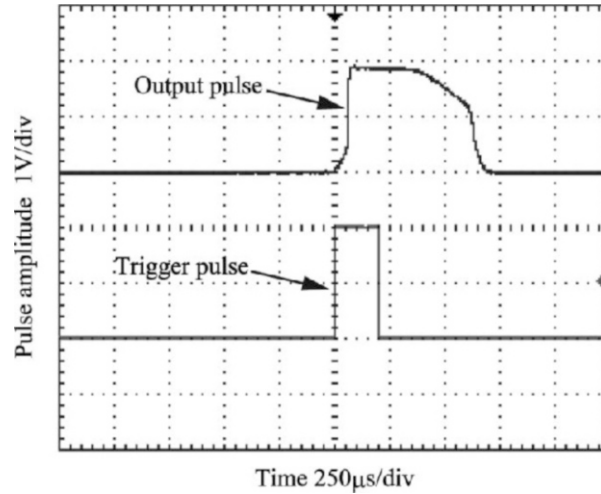


Figure 5.15. Example of the output waveform of the P-HNN

5.6. Conclusion

In this chapter, we presented the P-HNN which could output the PWM waveform of the PWM servo motor for robots. As a result, we generated the PWM waveforms of different pulse widths by varying the synaptic weights. The P-HNN realized the PWM servo motor control without using any software programs, or A/D converters. In addition, it is shown that the pulse waveform generation method of the P-HNN is same as for living organisms.

In the future, we will integrate the P-HNN circuit to the humanoid robot.

5.7. Acknowledgments

This research was supported by the JSPS KAKENHI (23760243), Nihon University College of Science and Technology Project Research, Nihon University Academic Research Grant (Total research, “11-002”). The fabrication of the silicon micro-robot was supported by the Research Center for Micro-Functional Devices, Nihon University.

5.8. Bibliography

- [DEL 80] DELCOMYN F., “Neural basis of rhythmic behavior in animals”, *Science*, vol. 210, pp. 492–498, 1980.
- [END 78] ENDO T., MORI S., “Mode analysis of a ring of a large number of mutually coupled van del Pol oscillators”, *IEEE Transactions on Circuits Systems*, vol. 25, no. 1, pp. 7–18, 1978.
- [GEN 95] GENTARO T., “A model of the neuro-musculo-skeletal system for human locomotion”, *Biological Cybernetics*, vol. 73, no. 2, pp. 97–111, 1995.
- [GIS 01] GISZTER S., MOXON K., RYBAK I., *et al.*, “Neurobiological and neurobotic approaches to control architectures for a humanoid motor system”, *Robotics and Autonomous Systems*, vol. 37, no. 2, pp. 219–235, 2001.
- [HAB 06] HABIB M.K., “Mechatronics engineering the evolution, the needs and the challenges”, *Proceedings of the 32nd Annual Conference of IEEE Industrial Electronics Society (IECON 2006)*, *IEEE*, pp. 4510–4515, 2006.
- [HAB 07] HABIB M.K., “Real time mapping and dynamic navigation for mobile robots”, *International Journal of Advanced Robotic Systems*, vol. 4, no. 3, pp. 323–338, 2007.
- [HAB 08] HABIB M.K., “Interdisciplinary mechatronics: problem solving, creative thinking and concurrent design synergy”, *International Journal of Mechatronics and Manufacturing Systems (IJMMS)*, vol. 1, no. 1, pp. 264–269, 2008.
- [HAT 06] HATA K., SAEKI K., SEKINE Y., “A pulse-type hardware CPG model for quadruped locomotion pattern”, *International Congress Series*, vol. 1291, March, pp. 157–160, 2006.
- [JIA 02] JIANG S., FUMIO N., “Neural locomotion controller design and implementation for humanoid robot HOAP-1”, *Proceedings of the 20th Annual Conference of the Robotics Society of Japan*, Osaka, Japan, 12–14 October 2002.
- [KAZ 98] KAZUO H., MASATO H., YUJI H., *et al.*, “The development of Honda humanoid robot”, *Proceedings of the 1998 IEEE International Conference on Robotics & Automation*, Leuven, Belgium, pp. 1321–1326, May 1998.
- [KEN 12a] KEN S., AKIHIRO M., KATSUTOSHI S., *et al.*, “Synchronization of coupled pulse-type hardware neuron models for CPG model”, *The Relevance of the Time Domain to Neural Network Models, Springer Series in Cognitive and Neural Systems*, vol. 3, pp. 117–133, 2012.
- [KEN 12b] KEN S., MINAMI T., YOSHIFUMI S., *et al.*, “Synchronization of coupled pulse-type hardware neuron models for CPG model”, *International Journal of Advanced Robotic Systems*, vol. 9, no. 226, pp. 1–6, 2012.

- [KIT 01] KITAJIMA H., YOSHINAGA T., AIHARA K., *et al.*, “Burst firing and bifurcation in chaotic neural networks with ring structure”, *International Journal of Bifurcation and Chaos*, vol. 11, no. 6, pp. 1631–1643, 2001.
- [KIY 87] KIYOTOSHI M., “Mechanisms of frequency and pattern control in the neural rhythm generators”, *Biological Cybernetics*, vol. 56, nos. 5–6, pp. 345–353, 1987.
- [KUN 03] KUNICHIKA T., TETSUYA Y., KAZUYUKI A., *et al.*, “Bifurcations in synaptically coupled Hodgkin-Huxley neurons with a periodic input”, *International Journal of Bifurcation and Chaos*, vol. 13, no. 3, pp. 653–666, 2003.
- [KUN 06] KUNICHIKA T., TETSUYA Y., HITOSHI I., *et al.*, “Bifurcations in a mathematical model for circadian oscillations of clock genes”, *Journal of Theoretical Biology*, vol. 239, no. 1, pp. 101–122, 2006.
- [MAT 02] MATSUOKA J., SEKINE Y., SAEKI K., *et al.*, “Analog hardware implementation of a mathematical model of an asynchronous chaotic neuron”, *IEICE Transactions on Fundamentals*, vol. E85-A, no. 2, pp. 216–221, February 2002.
- [MIC 02] MICHAEL A., *The Handbook of Brain Theory and Neural Networks*, 2nd ed., MIT Press, Cambridge, MA, 2002.
- [NAK 04] NAKABORA Y., SAEKI K., SEKINE Y., “Synchronization of coupled oscillators using pulse-type hardware neuron models with mutual coupling”, *Proceedings of the 2004 International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC 2004)*, pp. 8D2L-3-1–8D2L-3-4, 2004.
- [NAO 01] NAOMICHI O., NOBUTOSHI Y., “Generation of human bipedal locomotion by a bio-mimetic neuro-musculo-skeletal model”, *Biological Cybernetics*, vol. 84, no. 1, pp. 1–11, 2001.
- [RIA 02] RIADH Z., FUMIO N., “Recurrent neural network language for robot learning”, *Proceedings of the 20th Annual Conference of the Robotics Society of Japan*, Osaka, Japan, 12–14 October 2002.
- [SAE 03] SAEKI K., SEKINE Y., “CMOS implementation of neuron models for an artificial auditory neural network” *IEICE Transactions on Fundamentals*, vol. E86-A, no. 2, pp. 248–251, February 2003.
- [SAS 05] SASANO N., SAEKI K., SEKINE Y., “Short-term memory circuit using hardware ring neural networks”, *Artificial Life and Robotics*, vol. 9, no. 2, pp. 81–85, June 2005.
- [SHI 07] SHIGEKI T., TETSUSHI U., HIROSHI K., *et al.*, “Bifurcation analysis of current coupled BVP oscillators”, *International Journal of Bifurcation and Chaos*, vol. 17, no. 3, pp. 837–850, 2007.

- [SIM 00] SIMON F., KAREN A., ILYA R., *et al.*, “A neurobiological perspective on humanoid robot design”, *IEEE Intelligent Systems*, vol. 15, no. 4, pp. 64–69, 2000.
- [YAM 99] YAMAUCHI M., WADA M., NISHINO Y., *et al.*, “Wave propagation phenomena of phase states in oscillators coupled by inductors as a ladder”, *IEICE Transactions on Fundamentals*, vol. E82-A, no. 11, pp. 2592–2598, November 1999.
- [YAM 03] YAMAUCHI M., OKUDA M., NISHINO Y., *et al.*, “Analysis of phase-inversion waves in coupled oscillators synchronizing at in-and-anti-phase,” *IEICE Transactions on Fundamentals*, vol. E86-A, no. 7, pp. 1799–1806, July 2003.

Chapter 6

Parallel Wrists: Limb Types, Singularities and New Perspectives

This chapter reviews the criteria to identify the limb architectures suitable for parallel wrists (PWs) and the instantaneous kinematics of PWs. Special attention will be given to the use of the screw theory and of the displacement groups. The relationship between the limb architecture and the resulting PW mobility will be discussed. The main results reported in the literature about the singularity analysis of PWs will be summarized and the problem of characterizing the kinetostatic performances of PWs will be addressed. Eventually, innovative wrist architectures, recently proposed, will be presented.

6.1. Limb architectures and mobility analysis

PWs are parallel manipulators (PMs) with three degrees of freedom (DoF) where the end-effector motion can only be spherical. In PWs, the end-effector is connected to the frame through a number, n , of kinematic chains (limbs).

For a long time, there have been only two architectures of PWs diffusely studied in the literature: the 3-RRR wrist [GOS 89, GOS 94, GOS 95, ALI 94] and the S-3UPS wrist [INN 93, WOH 94]. The 3-RRR wrist (Figure 6.1) has three limbs of type RRR (R stands for revolute pair) with all

the revolute pair axes converging toward a single point of the frame (the center of the end-effector spherical motion). The S-3UPS wrist (Figure 6.2) has three limbs of type UPS (U, P and S stand for universal joint, prismatic pair and spherical pair, respectively) and a fourth limb that is constituted by a spherical pair that directly connects the end-effector and the frame.

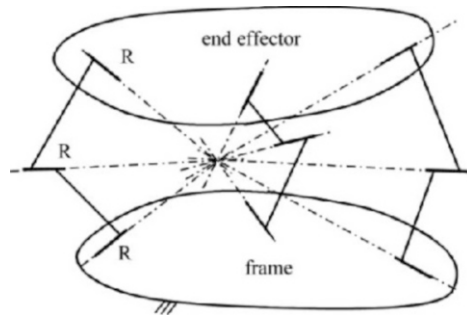


Figure 6.1. *The 3-RRR wrist*

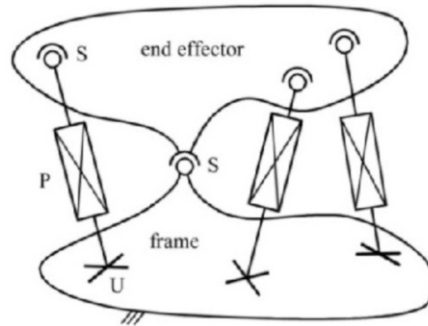


Figure 6.2. *The S-3UPS wrist*

Since 2000 [KAR 00], the interest in PWs has increased by following the increasing attention that the academic and industrial worlds paid to the PMs with lower mobility. This renewed interest in the PWs led to lot of new PW architectures appearing in the literature (see, for instance, [KAR 00, DIG 01a, DIG 01b, KAR 02, DIG 04a, KON 04a, KON 04b, KAR 04, KAR 06]). The analysis of this literature reveals that there are many techniques to generate limb architectures suitable for PWs. Such techniques can be collected into

three sets: (1) the techniques that use the screw theory, (2) the techniques that use group theory and (3) other techniques less general than screw and group theories.

In the following sections, these approaches to the problem of identifying PW limbs will be presented and discussed.

6.1.1. Use of the screw theory

6.1.1.1. Basic concepts

By introducing a reference point O (pole), a straight line, $(P, \mathbf{u})$, passing through a point P and with the direction of the unit vector $\mathbf{u}$, and two scalar coefficients, named q (signed magnitude) and p (pitch), whose product, $k (= q p)$, is always a finite real number, a screw can be geometrically defined as the following six-dimensional vector:

$$\hat{\$} = q \begin{bmatrix} \mathbf{u} \\ p\mathbf{u} + (\mathbf{P} - \mathbf{O}) \times \mathbf{u} \end{bmatrix} \quad [6.1]$$

The set of screws with the same pole, O , constitutes a vector space where the rules of the sum of two screws can be immediately deduced from the rules of the sum of two three-dimensional vectors.

According to the physical meaning given to q and to the line $(P, \mathbf{u})$, a screw can represent either a rigid body motion (in this case, the screw is named twist) or a system of forces (in this case, the screw is named wrench).

If q is the signed magnitude of a rigid body angular velocity, $\boldsymbol{\omega}$, that is equal to $q \mathbf{u}$, and the line $(P, \mathbf{u})$ is the instantaneous screw axis of an infinitesimal motion, the screw (twist) will represent the instantaneous motion of a rigid body since it has the following kinematic meaning:

$$\hat{\$}_t = \begin{bmatrix} \boldsymbol{\omega} \\ \mathbf{v}_O \end{bmatrix} \quad [6.2]$$

where $\mathbf{v}_O$ is the velocity that the pole O would have if it was fixed to the rigid body.

If q is the signed magnitude of the resultant force $\mathbf{F} (= q \mathbf{u})$ of a force system, and the line $(P, \mathbf{u})$ is Poinsot's central axis of the same force system,

the screw (wrench) will represent the force system since it has the following static meaning:

$$\hat{\mathbf{S}}_w = \begin{bmatrix} \mathbf{F} \\ \mathbf{M}_O \end{bmatrix} \quad [6.3]$$

where $\mathbf{M}_O$ is the resultant moment about the pole O of the force system.

The reciprocal product, $\hat{\mathbf{S}}_t \circ \hat{\mathbf{S}}_w$, between a twist, $\hat{\mathbf{S}}_t$, of the set of the twists with pole O (motion space) and a wrench, $\hat{\mathbf{S}}_w$, of the set of the wrenches with pole O (wrench space) is defined as follows:

$$\hat{\mathbf{S}}_t \circ \hat{\mathbf{S}}_w = \mathbf{F} \cdot \mathbf{v}_O + \mathbf{M}_O \cdot \boldsymbol{\omega} \quad [6.4]$$

The reciprocal product has the commutative and the distributive properties. It has the physical meaning of instantaneous virtual power that the system of forces, represented by $\hat{\mathbf{S}}_w$, introduces into a mechanical system that performs the instantaneous rigid motion represented by $\hat{\mathbf{S}}_t$.

A twist, $\hat{\mathbf{S}}_t$, is said to be reciprocal to a wrench, $\hat{\mathbf{S}}_w$, and vice versa if their reciprocal product, $\hat{\mathbf{S}}_t \circ \hat{\mathbf{S}}_w$, is equal to zero. Since, for an instantaneous motion compatible with the joint, the instantaneous virtual power introduced by the reaction forces of a passive frictionless joint is zero, it can be stated that, in a passive frictionless joint, the wrench that represents the reactions in the joint is reciprocal to the twist that represents the instantaneous relative motion between the links connected by the joint.

6.1.1.2. Identification of PW limbs

Let us consider a generic PM without actuators (i.e. all the joints are passive joints) where the end-effector is connected to the frame through n limbs (Figure 6.3), and call m_i the connectivity¹ of the i th limb ($i = 1, \dots, n$).

¹ The connectivity of a limb is the number of DoF that the end-effector would have, if it was connected to the frame only by that limb. If the limb is a serial kinematic chain, its connectivity will be equal to the number of joint variables of the limb. Such a number is also equal to the number of one DoF kinematic pairs, the limb would have, if all its multiple DoF pairs were transformed into a serial kinematic chain with one DoF pairs.

Hereafter, without losing generality, all the limbs will be assumed serial kinematic chains.

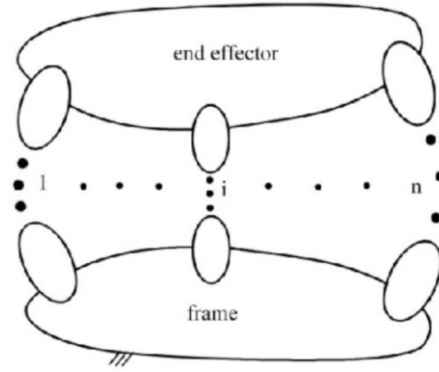


Figure 6.3. *Parallel manipulator with n limbs*

If only the i th limb connected the end-effector to the frame, the instantaneous virtual motion of the end-effector would be represented by a twist, $\hat{\mathcal{S}}_{t,i}$, that can be expressed as follows:

$$\hat{\mathcal{S}}_{t,i} = \sum_{j=1, m_i} \alpha_{ij} \hat{\mathcal{S}}_{t,ij} \quad [6.5]$$

where α_{ij} is the virtual rate of the j th joint variable of the i th limb; whereas $\hat{\mathcal{S}}_{t,ij}$ is the twist that represents the motion, the end-effector would have, if all the α_{ik} for $k = 1, \dots, (j - 1), (j + 1), \dots, m_i$ were equal to zero and α_{ij} was equal to 1. The explicit expressions of the $\hat{\mathcal{S}}_{t,ij}$ twists depend on the architecture and the configuration of the i th limb.

The force systems that the i th limb can exert on the end-effector are, of course, represented by wrenches, $\hat{\mathcal{S}}_{w,i}$, that are all reciprocal to $\hat{\mathcal{S}}_{t,i}$. The generic wrench $\hat{\mathcal{S}}_{w,i}$ can be represented through a base of six independent screws, $\hat{\mathcal{S}}_b$ for $b = 1, \dots, 6$, as follows:

$$\hat{\mathcal{S}}_{w,i} = \sum_{b=1,6} \beta_{ib} \hat{\mathcal{S}}_b \quad [6.6]$$

where β_{ib} for $b = 1, \dots, 6$ are scalar coefficients.

By using [6.5] and [6.6], the reciprocity of $\hat{\mathbf{S}}_{w,i}$ and $\hat{\mathbf{S}}_{t,i}$ yields the following analytic condition:

$$\sum_{j=1, m_i} \alpha_{ij} \left[\sum_{b=1,6} \beta_{ib} (\hat{\mathbf{S}}_b \circ \hat{\mathbf{S}}_{t,ij}) \right] = 0 \quad [6.7]$$

Since the α_{ij} coefficients can take any arbitrary value, condition [6.7] can be satisfied if and only if the following m_i conditions are matched:

$$\sum_{b=1,6} \beta_{ib} (\hat{\mathbf{S}}_b \circ \hat{\mathbf{S}}_{t,ij}) = 0 \quad j = 1, \dots, m_i \quad [6.8]$$

If the i th limb architecture is known, condition [6.8] is a system of m_i linear equations in the six β_{ib} coefficients. Provided that singular limb configurations are not considered, system [6.8] concludes that, if the connectivity m_i is greater than or equal to six, all the β_{ib} coefficients must be zero (i.e. the i th limb does not apply forces² to the end-effector). Therefore, only limbs with connectivity m_i less than six can actually constrain the end-effector motion.

According to [6.8], if m_i is less than six, there will be $(6 - m_i)$ coefficient β_{ib} that can be freely chosen in expression [6.6]. Therefore, the set of all the $\hat{\mathbf{S}}_{w,i}$ wrenches constitutes a screw system of order $(6 - m_i)$, which can be analytically expressed as follows:

$$\hat{\mathbf{S}}_{w,i} = \sum_{k=1, (6-m_i)} \gamma_{ik} \hat{\mathbf{S}}_{ik} \quad [6.9]$$

where γ_{ik} for $k = 1, \dots, (6 - m_i)$ are free coefficients, and the $\hat{\mathbf{S}}_{ik}$ are a set of $(6 - m_i)$ independent screws obtained by introducing the solutions of system [6.8] into expression [6.6].

The force systems that all the limbs apply to the end-effector are represented by the following wrench system:

$$\hat{\mathbf{S}}_e = \sum_{i=1, n} \sum_{k=1, (6-m_i)} \gamma_{ik} \hat{\mathbf{S}}_{ik} \quad [6.10]$$

² External loads (gravity, inertia forces, etc.) applied to the limb's links are not considered, and all the joints are considered passive in the analyses that bring to identify a limb architecture.

The PM is a PW if and only if the wrench system [6.10] coincides with the third-order screw system, $\hat{\mathbf{S}}_{F_p}$, that collects all the wrenches that represent a force, $\mathbf{F}_p$, with line of action passing through the center of the spherical motion, P. In order to make $\hat{\mathbf{S}}_e$ coincide with $\hat{\mathbf{S}}_{F_p}$, the following relationship must hold:

$$\hat{\mathbf{S}}_{ik} = \sum_{s=1,3} \delta_{ik,s} \left[\begin{array}{c} \mathbf{u}_s \\ (\mathbf{P}-\mathbf{O}) \times \mathbf{u}_s \end{array} \right] \quad i = 1, \dots, n; \quad k = 1, \dots, (6 - m_i) \quad [6.11]$$

where $\delta_{ik,s}$ for $s = 1, \dots, 3$ are suitable coefficients, and $\mathbf{u}_s$ for $s = 1, 2, 3$ are a set of three mutually orthogonal unit vectors.

In addition to [6.11], if the PW is non-overconstrained³, the following condition must hold too:

$$\sum_{i=1,n} (6 - m_i) = 3 \quad [6.12]$$

Provided that the limbs (redundant limbs) with m_i equal to six or greater than six are not considered; relationship [6.12] reveals that the non-overconstrained PWs without redundant limbs can be collected into two sets: (1) the PWs with three limbs with connectivity five, and (2) the PWs with two limbs: one with connectivity four and the other with connectivity five.

Once the connectivity and the topology of the i th limb are chosen, the generic expressions of the $\hat{\mathbf{S}}_{ik}$ can be computed by solving system [6.8]. Such expressions will contain the geometric parameters and the joint variables of the limb. And, by imposing that the computed $\hat{\mathbf{S}}_{ik}$ expressions satisfy relationship [6.11], the geometric conditions (i.e. the manufacturing and mounting conditions) that a limb with the chosen topology must satisfy to be used as a PW limb are found.

Such a technique is general and can be used to answer the questions such as “which are the PW limbs with only prismatic and revolute pairs?”. This

³ Parallel manipulators whose DoF number is greater than (equal to) the DoF number computed through the Grübler–Kutzbach rule are called overconstrained (non-overconstrained).

and other related problems have been investigated with this technique in [KON 04a].

6.1.2. Use of the group theory

6.1.2.1. Basic concepts

A group is a set, say $\{A\}$, with an associative binary operation, $\bullet$, so defined that (1) $a_1 \bullet a_2$ always exists and is an element of $\{A\}$ for any couple, a_1 and a_2 , of elements of $\{A\}$, (2) the identity element, e , of the operation $\bullet$ is an element of $\{A\}$ and (3) each $a \in \{A\}$ has an inverse element $a^{-1} \in \{A\}$.

The set of rigid body displacements (motions), $\{D\}$, is a six-dimensional group where the associative binary operation, $\bullet$, is the composition law of two displacements. The generic element of $\{D\}$ or of one out of its subgroups can be analytically represented by the screw identifying the finite or infinitesimal motion belonging to the subgroup. The dimension of a displacement subgroup is the number of independent scalar parameters that, in the analytic expression of the generic element's screw, must be varied to generate all the screws of the subgroup.

In addition to the identity subgroup, $\{E\}$, that corresponds to the absence of motion, $\{D\}$ contains 10 motion subgroups [HER 78, HER 99] with dimensions greater than zero and less than six:

(1) Subgroups of dimension 1:

(i) *Linear translation subgroup*, $\{T(\mathbf{u})\}$, that collects all the translations parallel to the unit vector $\mathbf{u}$. As many $\{T(\mathbf{u})\}$ as unit vectors, $\mathbf{u}$, can be defined. A prismatic pair with sliding direction parallel to $\mathbf{u}$ physically generates the motions of $\{T(\mathbf{u})\}$.

(ii) *Revolute subgroup*, $\{R(P, \mathbf{u})\}$, that collects all the rotations around an axis (rotation axis) passing through point P and parallel to the unit vector $\mathbf{u}$. As many $\{R(P, \mathbf{u})\}$ as rotation axes, $(P, \mathbf{u})$, can be defined. A revolute pair with rotation axis $(P, \mathbf{u})$ physically generates the motions of $\{R(P, \mathbf{u})\}$.

(iii) *Helical subgroup*, $\{H(P, \mathbf{u}, p)\}$, that collects all the helical motions with axis $(P, \mathbf{u})$ and finite pitch p that is different from zero and constant. As many $\{H(P, \mathbf{u}, p)\}$ as sets of helix parameters, $(P, \mathbf{u}, p)$, can be defined. A helical pair (hereafter denoted with H) with helix parameters $(P, \mathbf{u}, p)$ physically generates the motions of $\{H(P, \mathbf{u}, p)\}$.

(2) Subgroups of dimension 2:

(i) *Planar translation subgroup*, $\{T(\mathbf{u}_1, \mathbf{u}_2)\}$, that collects all the translations parallel to a plane perpendicular to $\mathbf{u}_1 \times \mathbf{u}_2$ where $\mathbf{u}_1$ and $\mathbf{u}_2$ are two orthogonal unit vectors. As many $\{T(\mathbf{u}_1, \mathbf{u}_2)\}$ as unit vectors $\mathbf{u}_1 \times \mathbf{u}_2$ can be defined.

(ii) *Cylindrical subgroup*, $\{C(P, \mathbf{u})\}$, that collects all the motions obtained by combining a rotation around a rotation axis $(P, \mathbf{u})$ and a translation parallel to the unit vector $\mathbf{u}$. As many $\{C(P, \mathbf{u})\}$ as $(P, \mathbf{u})$ axes can be defined. A cylindrical pair (hereafter denoted with C) with axis $(P, \mathbf{u})$ physically generates the motions of $\{C(P, \mathbf{u})\}$.

(3) Subgroups of dimension 3:

(i) *Spatial translation subgroup*, $\{T\}$, that collects all the spatial translations. Only one subgroup $\{T\}$ can be defined.

(ii) *Spherical subgroup*, $\{S(P)\}$, that collects all the spherical motions with center P. As many $\{S(P)\}$ as P points can be defined. A spherical pair (hereafter denoted with S) with center P physically generates the motions of $\{S(P)\}$.

(iii) *Planar subgroup*, $\{G(\mathbf{u}_1, \mathbf{u}_2)\}$, that collects all the planar motions with motion plane perpendicular to $\mathbf{u}_1 \times \mathbf{u}_2$ where $\mathbf{u}_1$ and $\mathbf{u}_2$ are two orthogonal unit vectors. As many $\{G(\mathbf{u}_1, \mathbf{u}_2)\}$ as unit vectors $\mathbf{u}_1 \times \mathbf{u}_2$ can be defined.

(iv) *Pseudo-planar subgroup*, $\{Y(\mathbf{u}_1, \mathbf{u}_2, p)\}$, that collects all the motions obtained by combining a planar translation belonging to $\{T(\mathbf{u}_1, \mathbf{u}_2)\}$ with a helical motion belonging to $\{H(P, \mathbf{u}_1 \times \mathbf{u}_2, p)\}$.

(4) Subgroups of dimension 4:

(i) *Schoenflies subgroup*, $\{X(\mathbf{u}_1, \mathbf{u}_2)\}$, that collects all the motions obtained by combining a planar translation belonging to $\{T(\mathbf{u}_1, \mathbf{u}_2)\}$ with a cylindrical motion belonging to $\{C(P, \mathbf{u}_1 \times \mathbf{u}_2)\}$.

A kinematic chain is called a mechanical bond when it connects one rigid body to another so that the relative motion between the two bodies is constrained. A mechanical bond is called mechanical generator when all the allowed relative motions between the two bodies belong to only one of the 10 subgroups of $\{D\}$.

6.1.2.2. Identification of PW limbs

If the end-effector was connected to the frame only by the i th limb, the generic element, l_i , of the set, $\{L_i\}$, of the displacements, the end-effector can perform, could always be expressed as follows:

$$l_i = a_{i1} \bullet a_{i2} \bullet \cdots \bullet a_{i(m_i-1)} \bullet a_{im_i} \quad [6.13]$$

where a_{ij} for $j = 1, \dots, m_i$ is a generic element of the set, $\{A_{ij}\}$, of the displacements that the j th joint⁴ of the i th limb allows.

The existence of relationship [6.13] between the elements of $\{L_i\}$ and the elements of the m_i sets, $\{A_{ij}\}$, can be reminded by introducing the following notation:

$$\{L_i\} = \{A_{i1}\} \bullet \{A_{i2}\} \bullet \cdots \bullet \{A_{i(m_i-1)}\} \bullet \{A_{im_i}\} \quad [6.14]$$

The set, $\{M\}$, of the displacements that the end-effector of a PM with n limbs can perform is a connected subset of the intersection of all the $\{L_i\}$ sets, that is:

$$\{M\} \subset \bigcap_{i=1,n} \{L_i\} \quad [6.15]$$

If $\{M\}$ is a subset of a spherical subgroup, $\{S(P)\}$, then the PM is a PW. Therefore, in order that the i th limb can be a PW limb, the existence of a connected subset, $\{S_i\}$, of $\{L_i\}$, which is a subset of the spherical subgroup, $\{S(P)\}$, too, is necessary. By imposing this condition, PW limbs can be identified; whereas, by imposing that $\{M\}$ is a subset of $\{S(P)\}$, PW architectures can be identified.

This scheme has been used in [KAR 00, KAR 02, KAR 04, KAR 06] to identify PW limbs and PW architectures.

6.1.3. Other approaches

Many other approaches have been used to demonstrate that specific PM topologies could be used as PW architectures. In this section, some of them are presented.

An approach different from the previous approaches has been used in [DIG 01a, DIG 01b]. This approach uses the velocity loop equations. The use of the velocity loop equations consists of exploiting the kinematic properties of the n limbs for writing n times both the end-effector angular velocity, ω ,

⁴ It has been assumed that the limb's joints are numbered from the frame to the end-effector.

and the velocity, $\mathbf{v}_P$, of the end-effector point, P , which is a candidate for being the center of the end-effector's spherical motion. By doing so, n expressions of the couple of vectors $(\boldsymbol{\omega}, \mathbf{v}_P)$ are obtained where the i th expression, $i = 1, \dots, n$, is a linear combination of the joint rates of the i th limb. The analysis of these $(\boldsymbol{\omega}, \mathbf{v}_P)$ expressions is sufficient to determine the geometric conditions that each limb has to satisfy in order to make (1) all the n expressions compatible, and (2) the velocity $\mathbf{v}_P$ equal to zero. Since this approach deduces geometric conditions by analyzing the instantaneous end-effector motion, the characteristics of the finite end-effector motion are stated by demonstrating that those conditions are sufficient to warrant an infinite sequence of instantaneous motion of the same type provided that no singular configuration is encountered.

The use of the velocity loop equations could be used as a general technique to find and/or enumerate PW topologies, but its implementation for this aim is much more complex than the implementation of the methods presented in the previous subsection. On the other hand, it gives many more pieces of information, about the instantaneous behavior of a specific architecture, than other methods, and it is specially recommended to find all the PW singularities since it considers all the joint rates of the PW.

Di Gregorio [DIG 04a] demonstrated that, if three non-aligned points of a rigid body are constrained to move on concentric spheres, the rigid body will be constrained to perform only a spherical motion. This demonstration can be used for finding PW architectures by assembling limbs that constrain an end-effector point to lie on a sphere. In [DIG 04a], it was used to propose the 3-RRS wrist (Figure 6.4).

Once a PW limb has been identified, other PW limbs can be generated. Indeed, by connecting two adjacent links of a PW limb, for instance the links joined by the j th joint, through a kinematic chain that, together with the j th joint, yields a single loop with one DoF, the added kinematic chain does not modify the type of motion that the end-effector can perform. Moreover, the configuration of the one DoF loop built around the j th joint is uniquely determined by the j th joint variable, and vice versa, the relative pose between the links is uniquely determined by the configuration of the added kinematic chain. Thus, the added chain is able to keep the mobility constraint, between the two links, due to the j th joint and, if the j th joint is replaced by that chain, the resulting limb architecture is a new PW limb. This procedure has been used in [KON 04b].

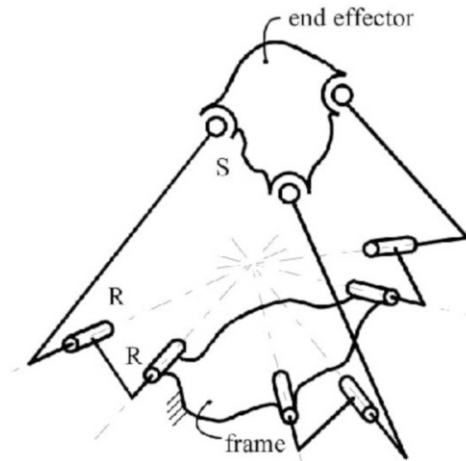


Figure 6.4. *The 3-RRS wrist*

Finally, new PW architectures can be obtained by adding limbs with connectivity 6 to already identified parallel or serial wrist architectures. The added limbs can contain actuated joints, the same as those already present in the original wrist architecture. Therefore, such a practice can be used to redistribute the actuators in a more convenient way. This principle can be seen as the one that generated the S-3UPS wrist (Figure 6.2) from a simple spherical pair that connects the end-effector to the frame.

6.1.4. Conclusion

Many architectures for PW limbs have been recently proposed by using either the screw theory or the group theory. Nevertheless, these are not the only tools that identify architectures suitable for being PW limbs. Moreover, once a limb architecture or a PW architecture has been identified, simple reasoning can be used to generate new limb architectures or new PW architectures.

6.2. Singularities and performance indices

Singularities are manipulator configurations where the relationship (input–output instantaneous relationship) between the rates of the actuated

joint variables and the characteristic vectors⁵ of the end-effector's instantaneous motion fails [GOS 90, MA 91, ZLA 95]. According to the input–output instantaneous relationship [GOS 90], they are of three types: (1) singularities of the inverse kinematic problem, (2) singularities of the direct kinematic problem and (3) singularities both of the inverse and of the direct kinematic problems.

Type 1 singularities occur when at least one of the input variable rates (actuated joint rates) is undetermined even though all the output variable rates (end-effector's motion characteristics $\{\boldsymbol{\omega}, \mathbf{v}_P\}$) are assigned. All the manipulator configurations where the end-effector reaches the border of the workspace are type 1 singularities, and finding type 1 singularities is one way to determine the workspace border. From a static point of view, in type 1 singularities, at least one component of output torque (force), applied to the end-effector, is equilibrated by the manipulator structure without applying any input torque (force) in the actuated joints.

Type 2 singularities occur when at least one component of end-effector's motion characteristics, $\{\boldsymbol{\omega}, \mathbf{v}_P\}$, is undetermined even though all the actuated joint rates are assigned. These singularities may be present only in PMs and fall inside the workspace. From a static point of view, in type 2 singularities, an (finite or infinitesimal) output torque (force), applied to the end-effector, needs at least one infinite input torque (force) in the actuated joints to be equilibrated. Since, long before the input torque (force) becomes infinite, the manipulator breaks down, type 2 singularities must be found during the design phase and avoided during operation.

This singularity classification has been extended in [ZLA 95] by taking into account the rates of the non-actuated joints.

In the literature [DIG 01a, DIG 01b, ZLA 01, DIG 02, ZLA 02, DIG 04b], the possibility of changing the type of motion the end-effector performs, in correspondence of particular type 2 singularities (constraint singularities) has been highlighted. Constraint singularities affect only PMs with lower mobility where the limbs' connectivity is greater than the manipulator's DoF. PWs are particular PMs with lower mobility. Therefore, PWs may have constraint singularities, that is configurations where the end-effector is no longer constrained to perform spherical motions.

⁵ The characteristic vectors of the instantaneous motion of a rigid body are the rigid body's angular velocity, $\boldsymbol{\omega}$, and the velocity, $\mathbf{v}_P$, of a rigid body point, P.

This section reviews the main results reported in the literature about the singularity analysis of PWs and addresses the problem of characterizing the kinetostatic performances of PWs.

6.2.1. Singularity analysis of PWs

The singularity analysis is the determination of all the singularities of a manipulator. In the configuration space of a manipulator (joint space or operational space), the geometric locus collecting all the points that identify manipulator singularities is named as the singularity locus.

According to the above-reported definitions, singularities are related to the input–output instantaneous relationship; thus, the implementation of the singularity analysis reduces itself to discuss such a relationship.

6.2.1.1. Analytical determination of the singularity loci

The input–output instantaneous relationship of a PM with lower mobility can be deduced by using the velocity loop equations [DIG 01a, DIG 01b, DIG 02]. Such a technique consists of two step: (1) the analytic calculus of a number of different expressions of the end-effector motion characteristics, $\{\omega, v_p\}$, equal to the limb number, say n (such a calculus considers each limb separately as it acted on the end-effector by itself); and (2) the elimination of all the rates of the non-actuated joint variables from the $6n$ scalar equations obtained in the previous step.

By doing so, the following input–output instantaneous relationship is obtained for a generic three DoF PM:

$$\begin{bmatrix} \mathbf{W} & \mathbf{V} \\ \mathbf{T} & \mathbf{U} \end{bmatrix} \begin{bmatrix} \omega \\ v_p \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{G} \end{bmatrix} \dot{\mathbf{q}} \quad [6.16]$$

where $\dot{\mathbf{q}}$ is the time derivative of the three-dimensional vector, $\mathbf{q}$, which collects the three actuated joint variables, q_i for $i = 1, 2, 3$. $\mathbf{H}$, $\mathbf{G}$, $\mathbf{T}$, $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{W}$ are 3×3 matrices that depend on the mechanism configuration (i.e. on $\mathbf{q}$). Such matrices can be seen as triplets of three-dimensional column vectors defined by the following formulas ($[\cdot]^T$ stands for transpose of $[\cdot]$):

$$\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3]; \mathbf{G} = [\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3]; \mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3]^T \quad [6.17a]$$

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3]^T; \mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]^T; \mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3]^T \quad [6.17b]$$

In a PW, if the center of the end-effector spherical motion is chosen as point P, the matrices $\mathbf{T}$ and $\mathbf{G}$ become null matrices, and relationship [6.16] becomes [DIG 01a]:

$$\begin{bmatrix} \mathbf{W} & \mathbf{V} \\ \mathbf{0}_{3 \times 3} & \mathbf{U} \end{bmatrix} \begin{bmatrix} \boldsymbol{\omega} \\ \mathbf{v}_P \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{0}_{3 \times 3} \end{bmatrix} \dot{\mathbf{q}} \quad [6.18]$$

where $\mathbf{0}_{3 \times 3}$ is the 3×3 null matrix. The homogeneity of the equations of system [6.18] allows the normalization of the $\mathbf{u}_i$ and $\mathbf{w}_i$ vectors for $i = 1, 2, 3$ (see definition [6.17b] of matrices $\mathbf{U}$ and $\mathbf{W}$). Therefore, without losing generality, it will be assumed that the $\mathbf{u}_i$ and $\mathbf{w}_i$ vectors are unit vectors.

The analysis of relationship [6.18] reveals that the end-effector is constrained to perform an infinitesimal (elementary) spherical motion if and only if matrix $\mathbf{U}$ is not singular. Indeed, only under this condition, the last three equations of system [6.18], that is:

$$\mathbf{U} \mathbf{v}_P = \mathbf{0} \quad [6.19]$$

have the only solution

$$\mathbf{v}_P = \mathbf{0} \quad [6.20]$$

that warranties the spherical motion.

Since matrix $\mathbf{U}$ is singular, if and only if its determinant is equal to zero, the values of $\mathbf{q}$ that satisfy the following singularity condition

$$\det(\mathbf{U}) = 0 \quad [6.21]$$

identify PW configurations where the type of motion of the end-effector may be non-spherical. Thus, such configurations are constraint singularities. Equation [6.21] is the analytical expression of the singularity locus that collects all the joint space points⁶ that correspond to constraint singularities of the PW.

If condition [6.21] is matched, system [6.19] has an infinite number of solutions for $\mathbf{v}_P$ (i.e. the velocity $\mathbf{v}_P$ is undetermined), even though the

⁶ The vector of the actuated-joint variables, $\mathbf{q}$, is the position vector that locates the points of the joint space.

actuated joint rates, $\dot{\mathbf{q}}$, are assigned. According to the above-reported singularity classification, this kinematic condition makes constraint singularities belong to the set of the type 2 singularities⁷.

By using the set theory, it can be demonstrated that constraint singularities may occur only in PM with lower mobility where all the limbs have a connectivity that is greater than the PM DoF (see [DIG 06] for details). Therefore, the PWs that have at least one limb with connectivity equal to three, like (see [GOS 95]) the 3-RRR (Figure 6.1) or the S-3UPS (Figure 6.2), have no constraint singularities.

Another criterion for avoiding constraint singularities is the use of PW architectures that have a constant and non-singular $\mathbf{U}$ matrix, like the 3-RRS wrist [DIG 04a] (Figure 6.4).

Out of constraint singularities, the input–output instantaneous relationship [6.18] reduces itself to the following relationship⁸:

$$\mathbf{W} \boldsymbol{\omega} = \mathbf{H} \dot{\mathbf{q}} \quad [6.22]$$

The analysis of [6.22] brings to the conclusion that: (1) if and only if matrix $\mathbf{W}$ is singular, the end-effector's angular velocity, $\boldsymbol{\omega}$, is not determined, even though the actuated joint rates, $\dot{\mathbf{q}}$, are assigned (i.e. a type 2 singularity occurs), and (2) if and only if matrix $\mathbf{H}$ is singular, the actuated joint rates, $\dot{\mathbf{q}}$, are not determined, even though the end-effector's angular velocity, $\boldsymbol{\omega}$, is assigned (i.e. a type 1 singularity occurs).

Because of the indetermination of $\boldsymbol{\omega}$, the type 2 singularities identified by condition (1) are also named rotation singularities [DIG 01a, DIG 01b, DIG 02, DIG 04b].

From an analytic point of view, the singularity condition (1) yields:

$$\det(\mathbf{W}) = 0 \quad [6.23]$$

⁷ Because of the indetermination of $\mathbf{v}_p$, the constraint singularities of PWs are also named translation singularities [DIG 01a, DIG 01b, DIG 02, DIG 04b].

⁸ Relationship [6.22] had been erroneously considered the complete input–output instantaneous relationship before the presence of constraint singularities was highlighted in the literature.

whereas the singularity condition (2) yields:

$$\det(\mathbf{H}) = 0 \quad [6.24]$$

Equations [6.23] and [6.24] are the analytical expressions of two singularity loci that collect all the joint space points that correspond, respectively, to the rotation singularities, and to the type 1 singularities.

6.2.1.2. Geometric interpretation of the singularity conditions

Definition [6.17b] of matrix $\mathbf{U}$ allows system [6.19] to be split into the following three scalar equations:

$$\mathbf{u}_i \cdot \mathbf{v}_p = 0 \quad i = 1, 2, 3 \quad [6.25]$$

From a kinematic point of view, the i th equation [6.25] shows that the velocity $\mathbf{v}_p$ has no component parallel to the vector $\mathbf{u}_i$, which, in other words, means that the end-effector point P cannot translate along the direction of the vector $\mathbf{u}_i$. Since the $\mathbf{u}_i$ vectors for $i = 1, 2, 3$ identify three directions along which the translation of P is forbidden, if the $\mathbf{u}_i$ vectors are linearly independent, point P cannot perform any elementary translation (i.e. condition [6.20] holds), otherwise it can translate along directions that are orthogonal to all the $\mathbf{u}_i$ vectors. The $\mathbf{u}_i$ vectors are linearly dependent if and only if they are all parallel to a unique plane. This geometric condition is analytically expressed as follows:

$$\mathbf{u}_1 \cdot \mathbf{u}_2 \times \mathbf{u}_3 = 0 \quad [6.26]$$

Because of definition [6.17b] of matrix $\mathbf{U}$, condition [6.26] coincides with condition [6.21], and it is its geometric counterpart.

Out of the constraint singularities, the end-effector motion is spherical, and system [6.22] is the input–output instantaneous relationship to be considered. Such a system is not singular, if and only if it states a one-to-one relationship between $\boldsymbol{\omega}$ and $\dot{\mathbf{q}}$. Since system [6.22] is linear and homogeneous with respect to $\boldsymbol{\omega}$ and $\dot{\mathbf{q}}$, $\boldsymbol{\omega}$ is determined (i.e. type 2 singularities do not occur), if and only if the only solution of system [6.22] for $\boldsymbol{\omega}$ is the null vector when the actuated joints are locked (i.e. $\dot{\mathbf{q}}$ is equal to zero). On the other hand, $\dot{\mathbf{q}}$ is determined (i.e. type 1 singularities do not occur), if and only if the only solution of system [6.22] for $\dot{\mathbf{q}}$ is the null vector when the end-effector is locked (i.e. $\boldsymbol{\omega}$ is equal to zero).

When the actuated joints are locked, definition [6.17b] of matrix $\mathbf{W}$ allows system [6.22] to be split into the following three scalar equations:

$$\mathbf{w}_i \cdot \boldsymbol{\omega} = 0 \quad i = 1, 2, 3 \quad [6.27]$$

From a kinematic point of view, the i th equation [6.27] shows that the end-effector's angular velocity $\boldsymbol{\omega}$ has no component parallel to the vector $\mathbf{w}_i$, which, in other words, means that the end-effector cannot rotate around an axis parallel to the vector $\mathbf{w}_i$ and passing through P. Since the $\mathbf{w}_i$ vectors for $i = 1, 2, 3$ identify three directions around which the end-effector rotation is forbidden, if the $\mathbf{w}_i$ vectors are linearly independent, the end-effector cannot perform any elementary rotation (i.e. $\boldsymbol{\omega}$ must be equal to the null vector), otherwise it can rotate around an axis passing through P and orthogonal to all the $\mathbf{w}_i$ vectors. The $\mathbf{w}_i$ vectors are linearly dependent if and only if they are all parallel to a unique plane. This geometric condition is analytically expressed as follows:

$$\mathbf{w}_1 \cdot \mathbf{w}_2 \times \mathbf{w}_3 = 0 \quad [6.28]$$

Because of definition [6.17b] of matrix $\mathbf{W}$, condition [6.28] coincides with condition [6.23], and it is its geometric counterpart.

By using definition [6.17a] of matrix $\mathbf{H}$, system [6.22] can be written as follows ($\dot{\mathbf{q}} \equiv [\dot{q}_1, \dot{q}_2, \dot{q}_3]^T$):

$$\mathbf{W} \boldsymbol{\omega} = \sum_{i=1,3} \mathbf{h}_i \dot{q}_i \quad [6.29]$$

Moreover, singularity condition [6.24], which identifies the type 1 singularities, can be rewritten as follows:

$$\mathbf{h}_1 \cdot \mathbf{h}_2 \times \mathbf{h}_3 = 0 \quad [6.30]$$

Condition [6.30] shows that a type 1 singularity occurs if and only if the three $\mathbf{h}_i$ vectors, for $i = 1, 2, 3$, are all parallel to a unique plane. When this condition occurs, the direction that is normal to all the $\mathbf{h}_i$ vectors is given by the cross product of any couple of non-parallel $\mathbf{h}_i$ vectors, say $\mathbf{n} \equiv \mathbf{h}_1 \times \mathbf{h}_2$. Thus, the dot product of the vector equation [6.29] by $\mathbf{n}$ yields, after rearrangements:

$$\boldsymbol{\omega}^T \mathbf{b} = 0 \quad [6.31]$$

where

$$\mathbf{b} = \mathbf{W}^T \mathbf{n} \quad [6.32]$$

Equation [6.31] shows that, when condition [6.30] is satisfied, the end-effector's angular velocity cannot assume any direction since it must be orthogonal to the vector $\mathbf{b}$. The existence of motion limitations on the end-effector identifies a manipulator configuration that is located at the borders of the workspace. Therefore, equation [6.30] is the analytic expression of the workspace borders in the manipulator's configuration space (joint space or operational space).

6.2.2. Kinetostatic performances

Every relationship of the instantaneous (first-order) kinematics has a static interpretation that can be determined through the virtual work principle. That is why the static interpretation of the input–output instantaneous relationship of a manipulator, sometimes, is referred to as kinetostatics [ANG 03].

6.2.2.1. Statics of PWs

By considering only the PW skeleton (i.e. the PW without generalized torques applied in the active joints), the virtual work principle yields the following relationship:

$$\mathbf{F} \cdot \mathbf{v}_P + \mathbf{M}_P \cdot \boldsymbol{\omega} = 0 \quad [6.33]$$

where $\mathbf{F}$ and $\mathbf{M}_P$ are resultant force and resultant moment about the pole P, respectively, of an external force system that is applied to the end-effector and is balanced by the reactions that all the limbs⁹ exert on the end-effector. $\mathbf{v}_P$ and $\boldsymbol{\omega}$ are any set of end-effector's motion characteristics compatible with the end-effector constraints (virtual motion characteristics).

Since the end-effector's constraints are analytically expressed by the instantaneous input–output relationship, the virtual motion characteristics are any set $\{\boldsymbol{\omega}, \mathbf{v}_P\}$ that satisfies relationship [6.18].

⁹ No external load (gravity, inertia forces, etc.) is applied to the limbs' links, all the kinematic pairs are non-actuated (passive) and frictionless.

According to [6.18], $\boldsymbol{\omega}$ is practically free to assume any value, whereas $\mathbf{v}_P$ must follow the vector equation [6.19]. Therefore, equation [6.33] can be satisfied (i.e. the manipulator skeleton is in equilibrium) if and only if $\mathbf{F}$ and $\mathbf{M}_P$ have the following analytic expressions:

$$\mathbf{M}_P = 0 \quad [6.34a]$$

$$\mathbf{F} = \sum_{i=1,3} \alpha_i \mathbf{u}_i \quad [6.34b]$$

where α_i , for $i = 1, 2, 3$, are three coefficients whose values must be computed through [6.34b] once the value of $\mathbf{F}$ is assigned.

Expressions [6.34] reveal that the PW skeleton can equilibrate a force system that is equivalent to a unique force with line of action passing through the center, P, of the end-effector spherical motion. Moreover, the detailed static analysis of any real case (see, for instance, [DIG 04b]) reveals that the loads applied on the limbs' links are proportional to the α_i coefficients.

By solving [6.34b] with respect to the α_i coefficients, the following explicit expressions of these coefficients result in:

$$\alpha_i = \frac{\mathbf{F} \cdot \mathbf{u}_j \times \mathbf{u}_k}{\mathbf{u}_i \cdot \mathbf{u}_j \times \mathbf{u}_k} \quad i, j, k \in \{1, 2, 3 \mid i \neq j, i \neq k, j \neq k\} \quad [6.35]$$

Since the absolute value of the denominator of expression [6.35] coincides with the absolute value of $\det(\mathbf{U})$, the more $\det(\mathbf{U})$ is near to zero (i.e. the more a PW configuration is near to a constraint singularity), the greater the absolute values of the α_i coefficients are (i.e. the greater the loads on the limbs' links are).

By considering the PW with generalized torques applied in the active joints, the virtual work principle yields the following relationship:

$$\mathbf{F} \cdot \mathbf{v}_P + \mathbf{M}_P \cdot \boldsymbol{\omega} = \boldsymbol{\tau} \cdot \dot{\mathbf{q}} \quad [6.36]$$

where $\boldsymbol{\tau}$ is a three-dimensional vector collecting the three generalized torques, τ_i for $i = 1, 2, 3$. $\dot{\mathbf{q}}$ is any vector of actuated joint rates compatible with the constraints (virtual rates of the actuated joint variables). $\mathbf{F}$, $\mathbf{M}_P$, $\mathbf{v}_P$ and $\boldsymbol{\omega}$ have the same meaning as they have in [6.33].

By choosing $\mathbf{F}$ according to [6.34b], the first term at the left-hand side of [6.36] is identically equal to zero. Moreover, out of constraint singularities, the input–output instantaneous relationship reduces itself to vector equation [6.29]. Thus, equation [6.36] is identically satisfied if and only if:

$$\boldsymbol{\tau} = \mathbf{H}^T \mathbf{W}^{-T} \mathbf{M}_p \quad [6.37a]$$

$$\mathbf{F} = \sum_{i=1,3} \alpha_i \mathbf{u}_i \quad [6.37b]$$

Since the explicit expression of $\mathbf{W}^{-T}$ is:

$$\mathbf{W}^{-T} = \frac{1}{\mathbf{w}_1 \cdot \mathbf{w}_2 \times \mathbf{w}_3} [\mathbf{w}_2 \times \mathbf{w}_3, \mathbf{w}_3 \times \mathbf{w}_1, \mathbf{w}_1 \times \mathbf{w}_2]^T \quad [6.38]$$

the right-hand side of [6.37a] has a common factor that is the reciprocal of $\det(\mathbf{W})$. As a result, the more $\det(\mathbf{W})$ is near to zero (i.e. the more a PW configuration is near to a rotation singularity), the greater the values of the generalized torques, τ_i for $i = 1, 2, 3$, are.

6.2.2.2. Performance indices

Characterizing the kinetostatic performances is important in order to compare different architectures during the design of a new manipulator.

The above-reported static analysis highlights that the force transmission from the end-effector to the links and to the actuators depends on the manipulator configuration. Thus, the indices devised to measure these manipulator performances must be configuration dependent, whereas the overall performances of the architecture can be measured by means of suitable averages on the workspace of the configuration-dependent indices.

In the literature, many indices were proposed to measure the kinetostatic performances of serial manipulators (see [ANG 03] for references), and, successively, adapted to PMs [GOS 91, GOS 95, ZAN 97].

In order to understand the static meaning of the indices used for PWs, the concept of manipulability measure introduced in [YOS 85] will be briefly recalled.

By using relationship [6.37a], a manipulating ability (manipulability) ellipsoid that identifies all the $\mathbf{M}_p$ values corresponding to a unit generalized-torque vector, $\boldsymbol{\tau}$, can be defined as follows:

$$\mathbf{x}^T \mathbf{J} \mathbf{J}^T \mathbf{x} = 1 \quad [6.39]$$

where

$$\mathbf{x} = \frac{\mathbf{M}_p}{|\boldsymbol{\tau}|} \quad [6.40a]$$

$$\mathbf{J} = \mathbf{W}^{-1} \mathbf{H} \quad [6.40b]$$

Since matrix $\mathbf{J}$ depends on the manipulator configuration, the shape of the ellipsoid [6.39] depends on the manipulator configuration. In particular, the lengths of the semi-axes are equal to the reciprocals of the absolute values of the $\mathbf{J}$ eigenvalues, the directions of the ellipsoid axes are given by the directions of the $\mathbf{J}$ eigenvectors, and the volume of the ellipsoid is proportional to the reciprocal of the product of all the $\mathbf{J}$ eigenvalues, which is equal to $\det(\mathbf{J})$.

The lengths of the semi-major and of the semi-minor axes of the manipulability ellipsoid give the maximum and the minimum values, respectively, of the ratio (mechanical advantage (MA)) between the magnitudes of $\mathbf{M}_p$ and $\boldsymbol{\tau}$. The manipulability ellipsoid highlights that, in general, the MA depends on the mechanism configuration and on the direction of $\mathbf{M}_p$. The manipulator configurations where the eigenvalues of the $\mathbf{J}$ matrix are all equal have an MA that does not depend on the direction of $\mathbf{M}_p$. Such configurations are named isotropic [ANG 03], and are characterized by the fact that their manipulability ellipsoid is a sphere. A PW with at least one isotropic configuration is named isotropic.

The distortion of the isotropic condition at a generic PW configuration can be measured by the following index (dexterity)¹⁰

$$\zeta(\mathbf{J}) = \frac{\lambda_{\min}}{\lambda_{\max}} \quad [6.41]$$

¹⁰ Definition [6.41] is a particular case of the following more general definition of dexterity [GOS 95] as inverse of the condition number of $\mathbf{J}$:

$$\zeta(\mathbf{J}) = \frac{1}{\|\mathbf{J}\| \|\mathbf{J}^{-1}\|} \quad [6.42]$$

where $\|\cdot\|$ denotes a norm of its matrix argument.

where $\lambda_{\min}$ and $\lambda_{\max}$ are the minimum and the maximum absolute values of the eigenvalues of matrix $\mathbf{J}$.

Moreover, at the parity of an ellipsoid shape (i.e. of dexterity), the greater the volume of the manipulability ellipsoid, the higher the MA values in any direction are. Such a kinetostatic property can be measured by the following “manipulability measure” [YOS 85]:

$$\mu(\mathbf{J}) = \sqrt{\det(\mathbf{J}\mathbf{J}^T)} \quad [6.43]$$

Since the determinant of the product of two square matrices is equal to the product of the determinants of the two matrices, definition [6.43] states that the “manipulability measure”, μ , is equal to the absolute value of $\det(\mathbf{J})$. And, definition [6.40b] makes it possible to state that $\det(\mathbf{J})$ is equal to the ratio between $\det(\mathbf{H})$ and $\det(\mathbf{W})$. As a result, definition [6.43] can be given in the following alternative way:

$$\mu(\mathbf{J}) = \frac{|\det(\mathbf{H})|}{|\det(\mathbf{W})|} \quad [6.44]$$

The “manipulability measure”, μ , ranges from zero to infinity and is inversely proportional to the ellipsoid volume. Definition [6.44] highlights that μ is equal to zero (i.e. the MA is equal to infinity) where $\det(\mathbf{H})$ is equal to zero (i.e. at the workspace borders (see singularity condition [6.24])), whereas it is equal to infinity (i.e. the MA is equal to zero) where $\det(\mathbf{W})$ is equal to zero (i.e. when a rotation singularity occurs (see singularity condition [6.23])). This result brings us to conclude that, from a static point of view, a PW must work far from the workspace borders and the rotation singularities. Finally, it is worth noting that, since the three $\mathbf{w}_i$ vectors (see definitions [6.17b]) are unit vectors, the PW configurations that are the farthest from rotation singularities are characterized by the fact that the absolute value of $\det(\mathbf{W})$ is equal to one (i.e. the three $\mathbf{w}_i$ vectors are mutually orthogonal (recall that $\det(\mathbf{W})$ is equal to $\mathbf{w}_1 \cdot \mathbf{w}_2 \times \mathbf{w}_3$)).

Both the dexterity and the manipulability measure characterize the force transmission, from the actuators to the end-effector, at a given PW configuration. In order to characterize the kinetostatic properties of a PW architecture, a global dexterity, ζ_{av} , and a global manipulability measure, μ_{av} , can be defined as follows:

$$\zeta_{av} = \frac{\int_Q \zeta(\mathbf{J}) dQ}{\int_Q dQ} \quad [6.45]$$

$$\mu_{av} = \frac{\int_Q \mu(\mathbf{J}) dQ}{\int_Q dQ} \quad [6.46]$$

where Q denotes the workspace of the PW.

So far, the defined indices take into account only the force transmission between end-effector and actuators (i.e. relationship [6.37a]). The force transmission between the end-effector and the PW skeleton is ruled by relationship [6.37b], which brings to relationship [6.35] for the α_i coefficients. Relationship [6.35] can be alternatively written as follows:

$$\boldsymbol{\alpha} = \mathbf{U}^{-T} \mathbf{F} \quad [6.47]$$

where

$$\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \alpha_3]^T \quad [6.48]$$

If, as usual [DIG 04b], the α_i coefficients are signed magnitudes of loads applied to the limbs' links, the resting ability (restability) of the PW skeleton can be characterized through the following restability ellipsoid:

$$\mathbf{y}^T \mathbf{U}^{-1} \mathbf{U}^{-T} \mathbf{y} = 1 \quad [6.49]$$

where

$$\mathbf{y} = \frac{\mathbf{F}}{|\boldsymbol{\alpha}|} \quad [6.50]$$

Since matrix $\mathbf{U}$ depends on the PW-skeleton configuration, the shape of the ellipsoid [6.49] depends on the PW-skeleton configuration. In particular, the lengths of the semi-axes are equal to the absolute values of the $\mathbf{U}$ eigenvalues, the directions of the ellipsoid axes are given by the directions of the $\mathbf{U}$ eigenvectors and the volume of the ellipsoid is proportional to the product of all the $\mathbf{U}$ eigenvalues, which is equal to $\det(\mathbf{U})$.

The lengths of the semi-major and of the semi-minor axes of the restability ellipsoid give the maximum and the minimum values, respectively, of the ratio (passive mechanical advantage (PMA)) between the magnitudes of $\mathbf{F}$ and $\boldsymbol{\alpha}$. The restability ellipsoid highlights that, in general, the PMA depends on the PW-skeleton configuration and on the direction of $\mathbf{F}$. The PW-skeleton configurations where the eigenvalues of the $\mathbf{U}$ matrix are all equal have a PMA that does not depend on the direction of $\mathbf{F}$. Such configurations uniformly distribute the force $\mathbf{F}$ among the links of the PW skeleton, and are characterized by the fact that their restability ellipsoid is a sphere.

The distortion of the uniform distribution condition at a generic configuration can be measured by the following redistribution index:

$$\xi(\mathbf{U}) = \frac{\eta_{\min}}{\eta_{\max}} \quad [6.51]$$

where $\eta_{\min}$ and $\eta_{\max}$ are the minimum and the maximum absolute values of the eigenvalues of matrix $\mathbf{U}$.

Moreover, at the parity of the ellipsoid shape (i.e. of ξ value), the greater the volume of the restability ellipsoid is, the higher the PMA values in any direction are. Such a kinetostatic property can be measured by the following restability index:

$$v(\mathbf{U}) = \frac{1}{\sqrt{\det(\mathbf{U}\mathbf{U}^T)}} \quad [6.52]$$

that can be alternatively defined as follows:

$$v(\mathbf{U}) = \frac{1}{|\det(\mathbf{U})|} \quad [6.53]$$

Since the three $\mathbf{u}_i$ vectors (see definitions [6.17b]) are unit vectors, the absolute value of $\det(\mathbf{U})$ (remind that $\det(\mathbf{U})$ is equal to $\mathbf{u}_1 \cdot \mathbf{u}_2 \times \mathbf{u}_3$) ranges from 0 at a constraint singularity (i.e. where singularity condition [6.21] is satisfied) to one at a configuration where the three $\mathbf{u}_i$ vectors are mutually orthogonal. As a result, the v index ranges from one to infinity and is inversely proportional to the ellipsoid volume. This result brings us to conclude that, from a static point of view, a PW skeleton must work far from

constraint singularities. Finally, it is worth noting that, the PW configurations that are the farthest from constraint singularities are characterized by the fact that the absolute value of $\det(\mathbf{U})$ is equal to one.

In order to globally characterize the kinetostatic properties of a PW skeleton, a global redistribution index, ξ_{av} , and a global restability index, v_{av} , can be defined as follows:

$$\xi_{av} = \frac{\int_Q \xi(\mathbf{U}) dQ}{\int_Q dQ} \quad [6.54]$$

$$v_{av} = \frac{\int_Q v(\mathbf{U}) dQ}{\int_Q dQ} \quad [6.55]$$

6.2.3. Conclusion

Instantaneous kinematics and statics are two sides of the same model: kinetostatics. Every relationship that is stated through the analysis of the instantaneous kinematics of a mechanism has a static meaning, too. Such a meaning can be deduced through the virtual work principle.

The input–output instantaneous relationship of a PM is the reference relationship for the introduction of the concept of singularity and for the classification of the PM singularities.

The input–output instantaneous relationship of a PW can be written in the general case, and the singularity conditions of PWs can be deduced both in analytical and geometrical form. The static meaning of the matrices that appear in the input–output instantaneous relationship is known together with the static interpretation of the singularity conditions.

Eventually, in the literature, kinetostatics has been exploited for proposing indices that characterize PWs' performances. Here, such indices have been discussed and integrated by proposing new indices that characterize the static efficiency of the PW skeleton.

6.3. New perspectives

Underactuated manipulators are able to position their end-effector in a workspace whose dimensionality¹¹ is greater than the number of actuators. Such a feature can be obtained by reducing the practicable paths that the end-effector can perform to move between two any end-effector poses of the workspace. Making the manipulator instantaneous DoF lower than its configuration DoF is a necessary condition to obtain a reduction of practicable paths. Since non-holonomic constraints are able to introduce this difference between instantaneous and configuration DoF, their introduction, in a manipulator architecture, is a way to obtain underactuated manipulators.

The roller–sphere contact is a non-holonomic constraint that constrains the sphere to rotate around axes that lie on the plane located by the roller axis and the sphere center. Rotations around axes that lie on a fixed plane are sufficient to make a rigid body assume any orientation. This fact was exploited by Stammers *et al.* [STA 91, STA 92] to conceive an underactuated wrist where two actuated rollers in contact with a sphere controlled the orientation (i.e. three configuration DoF) of an end-effector fixed to the sphere.

Grosch *et al.* [GRO 10] highlighted that, in “ordinary” (i.e. non-underactuated) manipulators, the substitution of passive spherical (S) pairs with as many non-holonomic spherical (nS) pairs yields underactuated manipulators that practically have the same workspace of the ordinary manipulators that have generated them. In [DIG 11], this substitution of S pairs with nS pairs was used to obtain 10 novel types of underactuated PWs from S-3UPS¹² wrists (Figure 6.2). The so-deduced underactuated PWs have the same workspace as the S-3UPS “ordinary” wrists that generated them, even though their hardware is simpler, but, due to their reduced instantaneous DoF, they cannot perform tracking tasks.

¹¹ In a manipulator, the workspace’s dimensionality is the minimum number of geometric parameters necessary to locate the end-effector in the operational space. If the manipulator is not redundant, such a number will coincide with the configuration (or finite) degrees of freedom (DoF) [ANG 03] of the manipulator, which is the minimum number of geometric parameters necessary to uniquely identify the manipulator configuration [ORE 08]. It may be different from the instantaneous DoF, also called velocity DoF [ANG 03], of the same manipulator.

¹² The underscore indicates an actuated pair.

The use of nS pairs was first proposed in [BEN 08]. Any S pair can be transformed into an nS pair (Figure 6.5) by introducing, in parallel with the S pair, a sphere–roller contact, whose sphere is a spherical shell, fixed to one of the two joined links and having the center coincident with the center of the S pair, whereas the roller is hinged on the other link (see [DIG 11, GRO 10] for details and constructive schemes). In Figure 6.5, the S pair is constituted by three revolute pairs in series, whose axes intersect one another at the same point; such a point is the center both of the S pair and of the nS pair. Because of the static friction, the sphere–roller contact forbids rotations around axes perpendicular to the plane (axes' plane (AP)), which the sphere center and the roller axis lie on. As a result, the relative motion of the two links joined by an nS pair can only be a rotation around an axis passing through the nS center and lying on the AP.

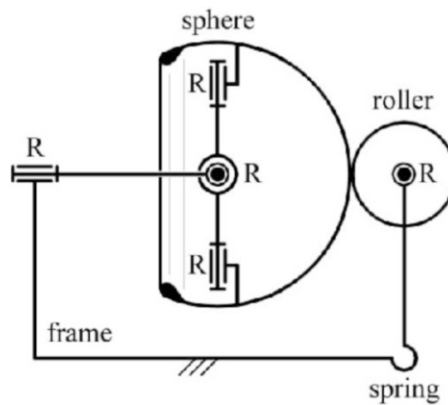


Figure 6.5. A manufacturing scheme for passive nS pairs, proposed in [DIG 11] (R stands for revolute pair): the nS center is the sphere center, and the axes' plane (AP) of this nS pair is the plane containing the roller axis and the sphere center

Three of the 10 underactuated PWs proposed in [DIG 11] contain only one nS pair: the (nS)-2SPU (Figure 6.6), the S-(nS)PU-SPU (Figure 6.7) and the S-(nS)PU-2SPU (Figure 6.8) underactuated PWs. These three PWs have been studied in depth [DIG 12a, DIG 12b, DIG 12c]. Such studies showed that their position analyses can be solved with simple formulas and that their kinetostatics is as simple as the kinetostatics of the S-3UPS that generated them. All these features brought us to conclude that they are a valid alternative to “ordinary” PWs when the wrist must not perform tracking tasks.

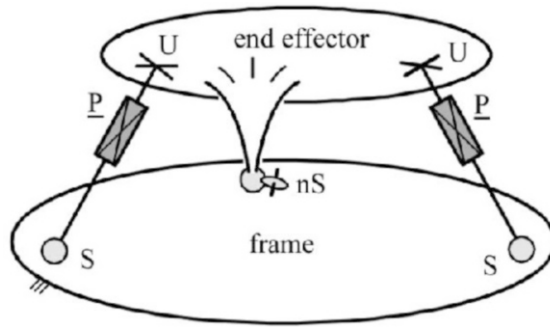


Figure 6.6. Underactuated parallel wrist of type $(nS)-2SPU$

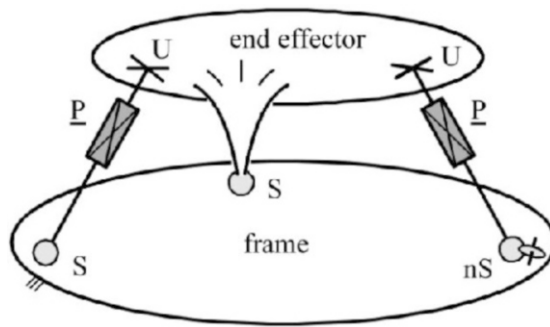


Figure 6.7. Underactuated parallel wrist of type $S-(nS)PU-SPU$

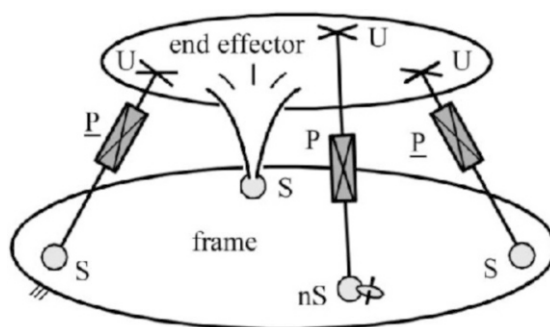


Figure 6.8. Underactuated parallel wrist of type $S-(nS)PU-2SPU$

6.4. Bibliography

- [ALI 94] ALIZADE R.I., TAGIYEV N.R., DUFFY J., “A forward and reverse displacement analysis of an in-parallel spherical manipulator”, *Mechanism and Machine Theory*, vol. 29, no. 1, pp. 125–137, 1994.
- [ANG 03] ANGELES J., *Fundamental of Robotic Mechanical Systems*, Springer-Verlag, New York, NY, 2003.
- [BEN 08] BEN-HORIN P., THOMAS F., “A nonholonomic 3-motor parallel robot”, in LENARCIC J., WENGER P. (eds), *Advances in Robot Kinematics: Analysis and Design*, Springer Science+Business Media B.V., Netherlands, pp. 111–118, 2008.
- [DIG 01a] DI GREGORIO R., “A new parallel wrist using only revolute pairs: the 3-RUU wrist”, *Robotica*, vol. 19, no. 3, pp. 305–309, 2001.
- [DIG 01b] DI GREGORIO R., “Kinematics of a new spherical parallel manipulator with three equal legs: the 3-URC wrist”, *Journal of Robotic Systems*, vol. 18, no. 5, pp. 213–219, 2001.
- [DIG 02] DI GREGORIO R., “A new family of spherical parallel manipulators”, *Robotica*, vol. 20, no. 4, pp. 353–358, 2002.
- [DIG 04a] DI GREGORIO R., “The 3-RRS wrist: a new, simple and non-overconstrained spherical parallel manipulator”, *ASME Journal of Mechanical Design*, vol. 126, no. 5, pp. 850–855, 2004.
- [DIG 04b] DI GREGORIO R., “Statics and singularity loci of the 3-UPU wrist”, *IEEE Transactions on Robotics*, vol. 20, no. 4, pp. 630–635, 2004.
- [DIG 06] DI GREGORIO R., “Parallel manipulators with lower mobility”, in MERDAN M. (ed.), *Industrial Robotics: Theory, Modelling and Control*, ARS International, Vienna, Austria, pp. 557–572, 2006.
- [DIG 11] DI GREGORIO R., “Under-actuated nonholonomic parallel wrists”, *Proceedings of the 13th World Congress in Mechanism and Machine Science*, IFToMM 2011, Guanajuato, Mexico, 19–25 June 2011, paper no. A12-264.
- [DIG 12a] DI GREGORIO R., “Kinematic analysis of the (nS)-2SPU underactuated parallel wrist”, *ASME Journal of Mechanisms and Robotics*, vol. 4, no. 3, 031006 (7 p.), 2012.
- [DIG 12b] DI GREGORIO R., “Position analysis and path planning of the S-(nS)PU-SPU and S-(nS)PU-2SPU underactuated wrists”, *ASME Journal of Mechanisms and Robotics*, vol. 4, no. 2, 021006 (6 p.), 2012.
- [DIG 12c] DI GREGORIO R., “Instantaneous kinematics and singularities of two types of under-actuated parallel wrists”, *Proceedings of the ASME 2012 11th Biennial Conference on Engineering Systems Design and Analysis*, ESDA 2012, Nantes, France, 2–4 July, paper no. ESDA2012-82084, 2012.

- [GOS 89] GOSSELIN C.M., ANGELES J., “The optimum kinematic design of a spherical three-degree-of-freedom parallel manipulator”, *ASME Journal of Mechanisms, Transmission and Automation in Design*, vol. 111, no. 2, pp. 202–207, 1989.
- [GOS 90] GOSSELIN C.M., ANGELES J., “Singularity analysis of closed-loop kinematic chains”, *IEEE Transactions on Robotics and Automation*, vol. 6, no. 3, pp. 281–290, 1990.
- [GOS 91] GOSSELIN C.M., ANGELES J., “A global performance index for the kinematic optimization of robotic manipulators”, *ASME Journal of Mechanical Design*, vol. 113, no. 3, pp. 220–226, 1991.
- [GOS 94] GOSSELIN C.M., SEFRIQUI J., RICHARD M.J., “On the direct kinematics of spherical three-degree-of-freedom parallel manipulators of general architecture”, *ASME Journal of Mechanical Design*, vol. 116, no. 2, pp. 594–598, 1994.
- [GOS 95] GOSSELIN C.M., PERREAULT L., VAILLANCOURT C., “Simulation and computer-aided kinematic design of three-degree-of-freedom spherical parallel manipulators”, *Journal of Robotic Systems*, vol. 12, no. 12, pp. 857–869, 1995.
- [GRO 10] GROSCH P., DI GREGORIO R., THOMAS F., “Generation of under-actuated manipulators with nonholonomic joints from ordinary manipulators”, *ASME Journal of Mechanisms and Robotics*, vol. 2, no. 1, 011005 (8 p.), 2010.
- [HER 78] HERVÉ J.M., “Analyse structurale des mécanismes par groupe des déplacements”, *Mechanism and Machine Theory*, vol. 13, no. 4, pp. 437–450, 1978.
- [HER 99] HERVÉ J.M., “The Lie group of rigid body displacements, a fundamental tool for mechanism design”, *Mechanism and Machine Theory*, vol. 34, no. 6, pp. 719–730, 1999.
- [INN 93] INNOCENTI C., PARENTI-CASTELLI V., “Echelon form solution of direct kinematics for the general fully-parallel spherical wrist”, *Mechanism and Machine Theory*, vol. 28, no. 4, pp. 553–561, 1993.
- [KAR 00] KAROUIA M., HERVÉ J.M., “A three-dof tripod for generating spherical rotation”, in LENARCIC J., STANISIC M.M. (eds), *Advances in Robot Kinematics*, Kluwer Academic Publishers, Netherlands, pp. 395–402, 2000.
- [KAR 02] KAROUIA M., HERVÉ J.M., “A family of novel orientational 3-dof parallel robots”, *Proceedings of the 14th CISM-IFTOMM Symposium on Robots and Manipulators (RoManSy'14)*, Springer, Udine, Italy, pp. 359–368, July 2002.
- [KAR 04] KAROUIA M., HERVÉ J.M., “New parallel wrists: special limbs with motion dependency”, in LENARCIC J., GALLETTI C. (eds), *On Advances in Robot Kinematics*, Kluwer Academic Publishers, Netherlands, pp. 371–380, 2004.
- [KAR 06] KAROUIA M., HERVÉ J.M., “Non-overconstrained 3-dof spherical parallel manipulators of type: 3-RCC, 3-CCR, 3-CRC”, *Robotica*, vol. 24, no. 1, pp. 85–94, 2006.

- [KON 04a] KONG X., GOSSELIN C.M., "Type synthesis of 3-DOF spherical parallel manipulators based on screw theory", *ASME Journal of Mechanical Design*, vol. 126, no. 1, pp. 101–108, 2004.
- [KON 04b] KONG X., GOSSELIN C.M., "Type synthesis of 3-DOF spherical parallel manipulators", *The International Journal of Robotics Research*, vol. 23, no. 3, pp. 237–245, 2004.
- [MA 91] MA O., ANGELES J., "Architecture singularities of platform manipulators", *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, Sacramento, CA, pp. 1542–1547, April 1991.
- [ORE 08] O'REILLY O.M., *Intermediate Dynamics for Engineers*, Cambridge University Press, New York, NY, 2008.
- [STA 91] STAMMERS C.W., PREST P.H., MOBLEY C.G., "The development of a versatile friction drive robot wrist", *Proceedings of the 8th World Congress on the Theory of Machines and Mechanisms*, Prague, Czech Republic, pp. 499–502, 26–31 August 1991.
- [STA 92] STAMMERS C.W., PREST P.H., MOBLEY C.G., "A friction drive robot wrist: electronic and control requirements", *Mechatronics*, vol. 2, no. 4, pp. 391–401, 1992.
- [WOH 94] WOHLHART K., "Displacement analysis of the general spherical Stewart platform", *Mechanism and Machine Theory*, vol. 29, no. 4, pp. 581–589, 1994.
- [YOS 85] YOSHIKAWA T., "Manipulability and redundancy control of robotic mechanisms", *Proceedings of 1985 IEEE International Conference on Robotics and Automation*, St. Louis, MO, pp. 1004–1009, 1985.
- [ZAN 97] ZANGANEH K.E., ANGELES J., "Kinematic isotropy and the optimum design of parallel manipulators", *The International Journal of Robotics Research*, vol. 16, no. 2, pp. 185–197, 1997.
- [ZLA 01] ZLATANOV D., BONEV I.A., GOSSELIN C.M., "Constraint singularities as configuration space singularities", available at www.parallemic.org/Reviews/Review008.html, 2001.
- [ZLA 02] ZLATANOV D., BONEV I.A., GOSSELIN C.M., "Constraint singularities of parallel mechanisms", *Proceedings of the 2002 IEEE International Conference on Robotics & Automation*, Washington, DC, pp. 496–502, May 2002.
- [ZLA 95] ZLATANOV D., FENTON R.G., BENHABIB B., "A unifying framework for classification and interpretation of mechanism singularities", *ASME Journal of Mechanical Design*, vol. 117, no. 4, pp. 566–572, 1995.

Chapter 7

A Robot-Assisted Rehabilitation System – RehabRoby

7.1. Introduction

Robot-assisted rehabilitation that can quantitatively monitor and adapt to patient progress and ensure consistency during rehabilitation has become an active research area. Robot-assisted rehabilitation systems have been developed for upper, lower, or both upper and lower extremities.

This chapter presents the development of an upper extremity robot-assisted rehabilitation system called RehabRoby. RehabRoby has been designed in such a way that (1) it can implement passive, active-assisted and resistive-assisted therapy modes, (2) it can be easily adjusted for people with different heights and arm lengths and (3) it can be used for both right and left arm rehabilitation.

Note that the control of RehabRoby in a desired and safe manner is an important issue. Impedance control [KRE 07, ROS 07, NEF 07, FRI 07, KOU 07], admittance control [LUM 06, LOU 03, ROS 07, NEF 07], position control [LUM 06] and force control [SAN 06] have previously been used in the control of the upper extremity robot-assisted rehabilitation systems. There is a human–robot interaction in the robot-assisted rehabilitation systems, which is an external effect that can cause changes in the dynamics of the

robotic systems. The changes in the dynamics of the robot may result in instability, which may affect the tracking performance. Therefore, a controller that is independent of the dynamic model of robot-assisted system is needed for RehabRoby to compensate changes in the dynamics of the robotic system [TSE 07]. Admittance control with an inner robust position control loop is used to control RehabRoby in a desired manner.

Note that it is also desirable for a patient to perform the rehabilitation task in a safe manner. A high-level controller, which is a decision-making mechanism, is designed to ensure safety during the execution of the rehabilitation task. The high-level controller presented in this chapter plays the role of a human supervisor (therapist) who would otherwise monitor the task and assess whether the rehabilitation task needs to be updated.

In this chapter, background of robot-assisted rehabilitation systems is presented in section 7.2. The control architecture of RehabRoby is given in section 7.3. In section 7.4, design specifications and hardware of RehabRoby are presented. Controllers designed for RehabRoby are described in section 7.5. Conclusion of the chapter and plans for future work are given in section 7.6.

7.2. Background

Rehabilitation robotics has been an active research area since 1985. Robot-supported therapy systems were first used in large-scale clinical tests in 1998, and until today several robot-assisted rehabilitation systems for upper extremities have been developed [KRE 04, TAK 05].

7.2.1. Robot-assisted rehabilitation systems for upper extremities

Existing robot-assisted rehabilitation systems for upper extremities either provide a therapy that focuses on multiple joint movements to perform activities of daily living (ADL) tasks or provide a therapy that focuses only on a single joint movement. End-effector-based or exoskeleton-type robots have been previously used to help patients to perform ADL tasks in therapy.

The end-effector-based robot is connected to the patient's limb from one point (hand or forearm). The exoskeleton-type robot is connected to the patient's arm from multiple points to resemble the kinematic structure of the arm. The technical rotation axes and the number of degrees of freedom (DoF) of the robot can be selected arbitrarily and independent of the human arm anatomy. Thus, the

mechanical design and construction of the end-effector-based robots are much easier [HES 05]. However, exoskeleton-type robots resemble the human arm anatomy, and the technical rotation axes of the robot must correspond to the rotation axis of the human joints [SAN 06]. Thus, the exoskeleton-type robots have a more complex mechanical design compared to end-effector-based robots. Additionally, the arm posture is fully determined, and each joint torque can be controlled separately in exoskeleton-type robots, which reduces control issue complexity, and hyperextensions can be avoided mechanically. MIT-Manus [KRE 07], mirror image movement enabler (MIME) [LUM 06], GENTLE/S [LOU 03] and NeRoBot [ROS 07] are previously developed end-effector-based robot-assisted rehabilitation systems. ARMin [NEF 07], T-WREX [HOU 07], Pneu-WREX [SAN 06], L-Exos [FRI 07] and Salford rehabilitation exoskeleton (SRE) [KOU 07] are existing exoskeleton-type robot-assisted rehabilitation systems.

MIT-Manus provides two-dimensional movements of the patient's hand [KRE 07]. The end-effector of MIT-Manus is the robot-mounted handle gripped by the patient, and forces and movements can be applied to the system using this handle. The MIME robot-assisted system has been developed with the cooperation of Stanford University and Veterans Affairs Palo Alto Health Care System [LUM 06]. Patients can achieve three-dimensional ADL movements during the therapy with MIME system. The MIME mechanism is composed of a PUMA 560, a six-DoF industrial robot manipulator, and a hand attachment in the end-effector. It is possible to execute passive, active and active-limited therapy methods using MIME. GENTLE/s is another end-effector-based robot, which consists of a three-DoF robot manipulator named Haptic Master and a virtual reality (VR) [LUM 06]. GENTLE/s allows patients to perform three-dimensional point-to-point movements. NeReBot has been designed as a three-DoF, wire-driven, end-effector-based robot for upper extremity rehabilitation [ROS 07]. There are three wires, which are connected to the patient's upper limbs through a spline in NeRoBot. The rehabilitation treatment based on the passive or active-assistive spatial motion of the limb is provided controlling the lengths of the wires driven by electric motors.

ARMin, which has been designed for arm therapy is an exoskeleton robot equipped with position and force sensors [NEF 07]. ARMin has four active and two passive DoF to allow elbow flexion/extension and spatial shoulder movements. Later, a second version of ARMin, called ARMin II, has been developed [NEF 07]. The mechanical structure, actuators and sensors of the ARMin have been optimized for the applications of impedance and

admittance control for ARMin II. Three therapy modes, which are passive mobilization, game therapy and task-oriented training, can be applied to patients with ARMin II. A new ergonomic shoulder actuation principle and its implementation of ARMin II have been developed, which is called ARMin III [NEF 07]. Three actuated DoF for the shoulder and one for the elbow joint are included in ARMin III. Actuated lower arm pronation/supination and wrist flexion/extension are made available with the additional module in ARMin III. Currently, ARMin III is in use for clinical evaluation in hospitals in Switzerland and the United States. The T-WREX (therapy Wilmington robotic exoskeleton) is a passive, five-DoF and body-powered device with no actuators exoskeleton. T-WREX has been designed to enable patients with significant arm weakness to achieve intense movement training without the expense of a physiotherapist [HOU 07]. It provides a large three-dimensional workspace that is approximately 66% of the natural workspace of the arm in the vertical plane and 72% in the horizontal plane. Weak patients can move their affected arm easily with the support provided against the gravity. Pneu-WREX is a robotic version of T-WREX that can apply a wide range of forces to the arm during upper extremity movements using pneumatic actuators [SAN 06]. L-Exos (light exoskeleton) is an exoskeleton robot with force feedback that has five DoF, four of which are actuated, and it can apply a controllable force up to 100 N at the center of the patient's hand palm [FRI 07]. The results of the clinical trials demonstrate that L-Exos can be used for robotic arm rehabilitation therapy when it is integrated with a VR system. The SRE is a gravity-compensated arm rehabilitation exoskeleton robot with seven DoF [KOU 07]. Three of these DoF are located at the shoulder for flexion/extension, abduction/adduction and lateral/medial rotation. Two are at the elbow for flexion/extension and pronation/supination of the forearm. The other two provide flexion/extension and abduction/adduction located at the wrist. Pneumatic actuation techniques that provide accurate position and force-controlled paths, compliance and a high level of inherent safety are used in the design of the exoskeleton.

7.2.2. Controllers of robot-assisted rehabilitation systems for upper extremities

Control of a robot-assisted rehabilitation system is an important issue to complete the rehabilitation task in a desired and safe manner. MIT-Manus uses impedance controller to support the motion of the hand to the target position. MIT-Manus is back-drivable with low inertia and friction [KRE 07].

Thus, it is possible to complete the rehabilitation task in a safe manner during the interaction between the patient and the robot. Force and position sensors are used to feed the impedance controller. Position and admittance control strategies are implemented with the six-DoF force–torque sensor and position sensors in MIME to execute four different control modes (passive, active-assisted, active constrained and bilateral modes) [LUM 06]. GENTLE/s provides assistance to the patients to move to the target points along the predefined trajectories using the admittance control [LOU 03]. Switching proportional-integral-derivative (PID) control has been used for the position control of NeReBot [ROS 07]. Impedance and admittance control techniques have been used for the ARMin robot-assisted rehabilitation systems [NEF 07]. Nonlinear force control and passive counter balancing techniques have been used for Pneu-Wrex [SAN 06]. Impedance control has been used for L-Exos and SRE [FRI 07, KOU 07].

7.3. Control architecture

A control architecture, which is composed of hardware and software components to complete the rehabilitation tasks in a desired and safe manner, has been developed for robot-assisted rehabilitation system RehabRoby [ÖZK 11a, ÖZK 11b, ÖZK 11c, ÖZK 12]. The control architecture consists of the rehabilitation robot (RehabRoby), low-level and high-level controllers, and a sensory information module. The block diagram of the control architecture is illustrated in Figure 7.1.

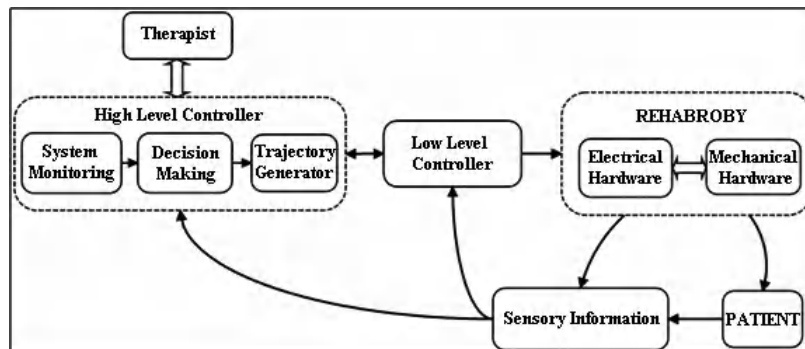


Figure 7.1. Control architecture of RehabRoby

7.4. RehabRoby

Upper extremity movement characteristics of a human are evaluated during the design phase of RehabRoby. Orthopedic society has provided a standard terminology, which is based on a consensus on three items, to describe upper extremity movements and movement limits for common clinical examination [LIV 65]. First, all positions are referenced on the anatomical posture defined as zero positions of the joint. Second, joint positions are measured in one of the three (orthogonal) planes (sagittal, frontal or transversal) or around the longitudinal axis (rotation). Finally, the degrees of motion are recorded as the deviation from the reference position in either direction from the anatomical position in a standardized format.

RehabRoby has been designed to provide basic upper extremity rehabilitation movements (extension, flexion, abduction, adduction, rotation, pronation and supination) and also a combination of these movements that are necessary for ADL. RehabRoby's motion axes are given in Figure 7.2. The range of motion (ROM), joint torques, velocities and accelerations for RehabRoby have been determined using the measurements of the movements of a healthy subject during two ADL tasks [FAS 04, OLD 07]. Higher joint torque values given in [FAS 04] and [OLD 07] have been selected to assure that RehabRoby is strong enough to overcome resistance from the human against movements due to spasms and other complications that are difficult to model. These joint torque values are used to select the proper combination of motors and gear units for each joint of RehabRoby. Motion specifications of RehabRoby are given in Table 7.1.

Axis	ROM (deg)	Maximum Torque (Nm)	Maximum Velocity (deg/s)
θ_1	-135° to 45°	34.9	332.28
θ_2	-135° to 45°	23.35	447.6
θ_3	-90° to 90°	90.44	78.16
θ_4	-90° to 30°	34.9	332.28
θ_5	-90° to 90°	53.4	72.44
θ_6	-50° to 79°	8.5	483

Table 7.1. Motion specifications of RehabRoby

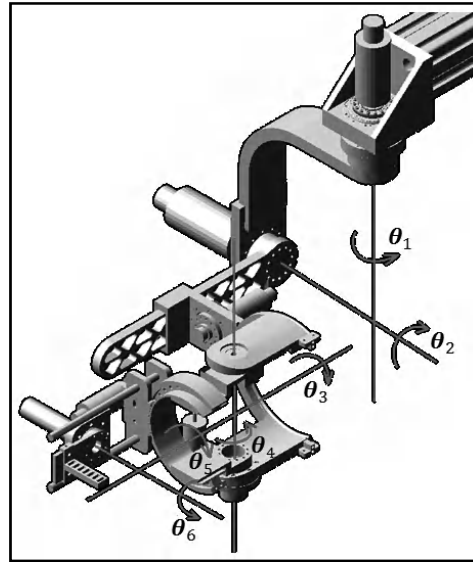


Figure 7.2. RehabRoby's axes (θ_1 : horizontal abduction/adduction of shoulder rotation, θ_2 : shoulder flexion/extension elevation, θ_3 : internal and external rotation of shoulder, θ_4 : elbow flexion/extension, θ_5 : lower arm elbow pronation/supination and θ_6 : wrist flexion/extension)

Maxon's brushed DC motors, EPOS model drivers (Maxon Motor AG, Switzerland) and gear units of Harmonic Drive (Harmonic Drive Inc., Japan) have been selected for the actuation of the joints of RehabRoby. There is a coupling between flexion/extension and abduction/adduction of shoulder axis. The position of the horizontal shoulder rotation angle, which is defined as θ_1 in Figure 7.2, determines the separation of the shoulder movements. When θ_1 is 0° , θ_2 , which represents the position of the flexion/extension and abduction/adduction of shoulder axis as shown in Figure 7.2, is responsible for the flexion/extension of shoulder. When θ_1 is 90° , θ_2 is responsible for the abduction/adduction of shoulder.

An arm splint is designed and attached to RehabRoby as shown in Figure 7.3. It has humeral and forearm thermoplastic supports with Velcro straps and a single axis free elbow joint. A thermoplastic inner layer covered by soft material (plastazote) is used due to the differences in the size of the subjects' arms. Thus, the total contact between the arm and the splint can be achieved to eliminate the loss of movement during the execution of the task.



Figure 7.3. *The RehabRoby system with arm splint and emergency buttons*

Ensuring safety of the subject is an important issue when designing a robot-assisted rehabilitation system. Thus, in case of emergency situations, the physiotherapist can press an emergency stop button to stop the RehabRoby (Figure 7.3). The motor drivers of RehabRoby can be disabled separately or together by pressing enable/disable buttons without disconnecting the energy of the RehabRoby. The power of the system is supported by an uninterruptible power supply. Thus, there is no power loss, and RehabRoby will not collapse at any time. Additionally, the rotation angle and angular velocity of each joint of RehabRoby are monitored by the high-level controller described in section 7.5.

RehabRoby has been designed in such a way that it can be easily adjustable for people of different heights and arm lengths. Anthropometric approaches have been used in the design of RehabRoby. The link lengths of RehabRoby are based on the arm lengths of 2,100 people in 14 cities in Turkey [GUL 07]. The adjustable link lengths and height of RehabRoby are shown in Figure 7.4. L_1 , which is the adjustable upper arm length value, varies from 260 to 400 mm. L_2 , which is the adjustable lower arm length

value, varies from 200 to 300 mm. Additionally, RehabRoby's height (L_3) can be adjusted for each subject using a screw shaft mechanism that can be manually operated using a wheel.

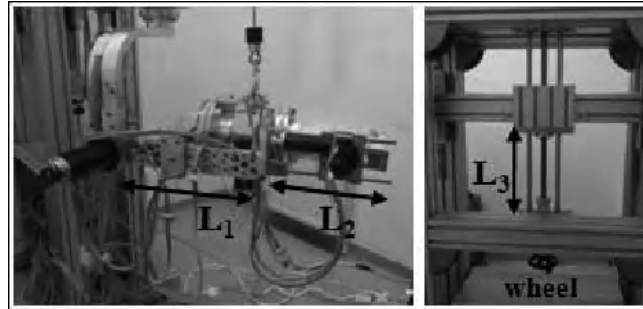


Figure 7.4. Adjustable link lengths and height of RehabRoby

RehabRoby is integrated with a counterweight mechanism as illustrated in Figure 7.5 to reduce the effect of gravity to help subjects to flex their shoulders easily. Note that the counterweight system is designed in such a way that it does not interfere with the subject's workspace.



Figure 7.5. RehabRoby with a counterweight system

RehabRoby can be used for both right and left arm rehabilitation. It can be translated from right arm use to left arm use with the following steps: (1) RehabRoby is rotated 90° about θ_2 ; (2) then, it is rotated 180° about θ_1 ; and (3) finally, it is rotated -90° about θ_2 .

RehabRoby has an interface with Matlab[®] Simulink/Realtime Workshop to allow fast and easy system development. Humusoft Mf624 model (Humusoft Inc., Czech Republic) data acquisition board is selected to provide real-time communication between the computer and other electrical hardware. Humusoft Mf624 data acquisition board is compatible with Real-Time Windows Target toolbox of Matlab[®]/Simulink. Digital incremental quadrature encoders are coupled with brushed DC motors for joint position measurement. Five of the six encoders have resolutions of 500 counts/turn, and one of them has a resolution of 1,000 counts/turn. Kistler's press force sensors (Kistler Holding AG, Winterthur, Switzerland), which are quite small in size, are selected to measure contact forces between the subject and RehabRoby. Two force sensors are placed in the inner surface of the thermoplastic molded plate attached dorsally to the forearm splint via Velcro straps in such a way that their measurement axes are perpendicular to each other. The placement of the force sensors is illustrated in Figure 7.6. One of the force sensors is used to measure the applied force during the elbow flexion movement. The other one measures the applied force during the shoulder flexion movement. Digital encoder data of motors and analog force data from the force sensors are received through the data acquisition board with a sampling frequency of 500 Hz.

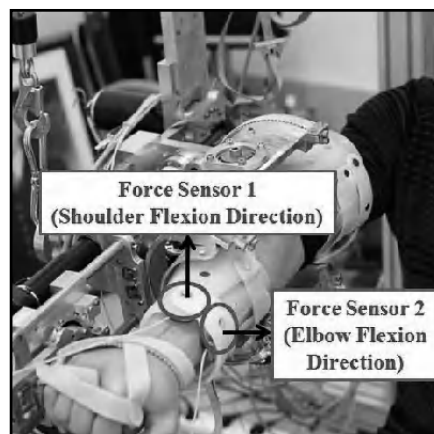


Figure 7.6. *Placement of the force sensors*

The closed-loop data flow in the control hardware occurs between the computer, data acquisition board, microcontroller circuits, motor drivers and motors with encoders. The control inputs, which are the current reference values of the motors of RehabRoby, are transmitted to the microcontroller circuits through analog outputs of the data acquisition board Humusoft Mf624 with a sampling frequency of 500 Hz. The incoming analog data are converted into digital data and transmitted to the motor drivers using RS232 serial bus with a baud rate of 115,200 by programmable interface controller (PIC) microcontrollers (Microchip Technology Inc., AZ, USA) in the microcontroller circuits. Here, microcontroller circuits are used because four of the six motor drivers of RehabRoby have no analog reference inputs. Analog to digital conversion and serial transmission are completed within 2 ms. Motor drivers send the reference current values to the motors using a simple current control algorithm to equalize the current values of the motors with the reference ones. Angular changes in the axes are measured by digital encoders coupled with the motors of RehabRoby and transmitted to the Matlab®/Simulink model in the computer as feedback through encoder inputs of the data acquisition board Humusoft Mf624. The block diagram of the general data flow in the hardware is shown in Figure 7.7.

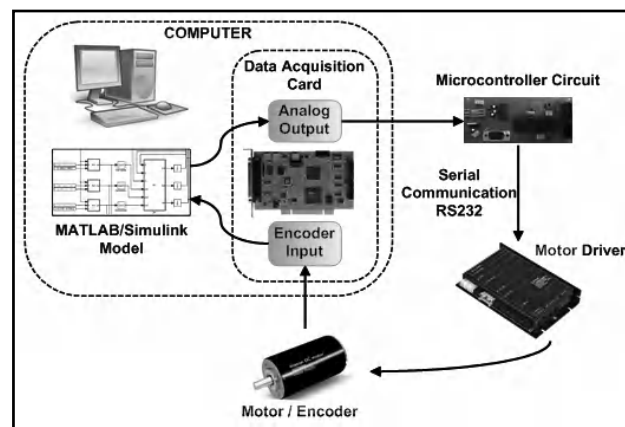


Figure 7.7. General data flow in the control hardware

7.5. Controllers of RehabRoby

The control structure of RehabRoby consists of low-level and high-level controllers. Admittance control with inner robust position control loop is used to control RehabRoby to provide assistance to the patients. The position

control of the joints of RehabRoby is provided by a robust controller with a Kalman filter-based disturbance estimator to minimize the effects of the uncertainties in the dynamics of RehabRoby because of its complex structure [STA 99]. The interaction forces between the subjects and RehabRoby are controlled using the admittance control technique. Additionally, a high-level controller is designed as a decision-making mechanism of RehabRoby using hybrid system modeling technique to monitor the task and assess whether the rehabilitation task or any parameter in the low-level controller needs to be updated.

7.5.1. Low-level controller

The low-level controller is responsible for providing necessary motion to RehabRoby. Therefore, patients can complete the rehabilitation tasks in a desired manner. An admittance control with an inner robust position control loop is used as the low-level controller of RehabRoby.

The admittance control method, which has low back drivability, high inertia and reliable position and force/torque information, is a good choice for control applications of the robotic systems [NEF 07]. Moreover, the position and torque sensors of RehabRoby have high resolutions, so admittance control could be a good choice. Because RehabRoby has complex and uncertain inner dynamics and it is sensitive to external forces during the human–robot interaction, a simple PID or model-based position control technique may not be enough. Hence, a robust position controller is used in the inner loop of the admittance controller. The effects of the parametric uncertainties in the dynamic model and the external additive disturbances are compensated with an equivalent disturbance estimator in the robust position controller. Various methods have been previously used to estimate the disturbance in the position control of robotic systems such as an adaptive hierarchical fuzzy algorithm and a model-based disturbance attenuation [EMA 04, CHO 03]. A discrete Kalman filter-based disturbance estimator [STA 99, JUN 98], which is a well-known technique for processing noisy discrete measurements and high-accuracy estimation of the unknown states and parameters, is used in this study. To our knowledge, admittance control with inner robust position control loop has not been used to control robot-assisted rehabilitation systems before.

The general structure of the proposed low-level controller for RehabRoby is given in Figure 7.8. The force that is applied by the subject during the execution of the task is measured using the force sensor, and this value is

then converted to torque using Jacobian matrix. The torque value is then passed through an admittance filter, which is used to define characteristics of the motion of the RehabRoby against the applied forces, to generate the reference motion for the robust position controller [JUN 98]. The reference motion is then tracked with a robust position control that consists of a linear Kalman filter-based disturbance estimator [STA 99, SAL 95].

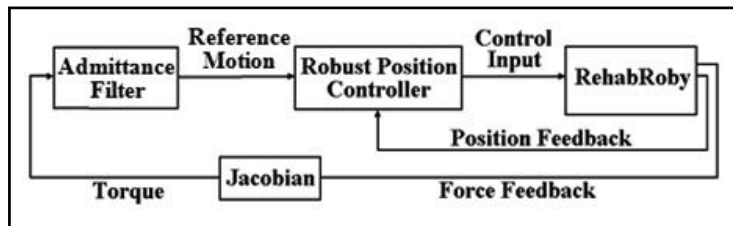


Figure 7.8. Block diagram of the low-level controller of RehabRoby

7.5.2. High-level controller

The high-level controller is the decision-making mechanism of RehabRoby. It decides necessary changes by analyzing information that comes from the sensory information module or the therapist. The high-level controller plays the role of a human supervisor (therapist) who would otherwise monitor the task and assess whether the task needs to be updated. A hybrid system modeling technique is used to design the high-level controller because it is easy to add new rules related to a rehabilitation task using this technique.

The block diagram of a high-level controller is illustrated in Figure 7.9. Initially, the states of the high-level controller are defined. When task execution commences, the starting and final positions of the joint angles of RehabRoby are initialized in the initialization state. Passive state (mode = 0) (passive mode), active state (mode = 1) (active-assisted mode) or admittance control state (mode = 2) (resistive-assisted mode) become active based on the mode selected by the therapist. The rehabilitation task is performed only in the passive state in which RehabRoby is responsible to help the subject complete the task when he/she is passive in the passive mode. The subject's motion is checked periodically in the active-assisted and resistive-assisted modes. If the subject's movement, which is measured as (θ) of RehabRoby, is out of limits $(\theta \geq |\epsilon|)$, then the position control state becomes active. When

the position control state is active, the RehabRoby provides assistance to the subject's motion until the subject's movement is in the desired motion range. When the subject's movement is in the range of limits ($\theta < |\varepsilon|$), the state that is active before entering the position control state becomes active again. In any state, safety conditions of RehabRoby are checked periodically, and if any unsafe situation occurs ($e = 1$), then the emergency stop state becomes active, and the execution of the task stops.

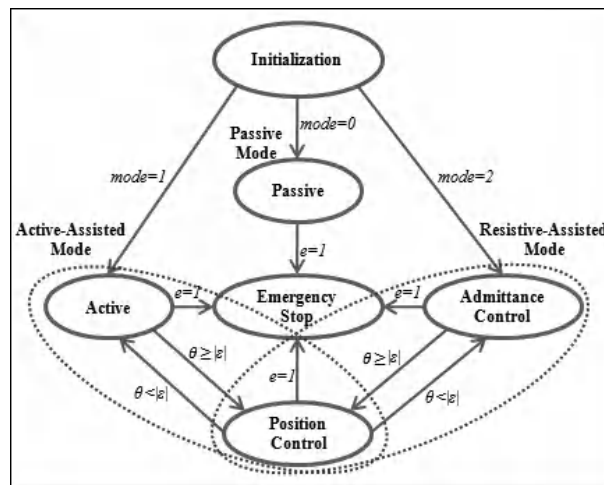


Figure 7.9. Block diagram of the high-level controller of RehabRoby

7.6. Concluding remarks

An exoskeleton-type upper extremity robot-assisted rehabilitation system, called RehabRoby, has been developed. RehabRoby is adaptable for patients of both genders, is adjustable for people with different arm lengths, and is usable for both right and left arms.

A control architecture which consists of a high-level controller and a low-level controller has been developed for RehabRoby. Low-level controllers can provide the necessary motion to RehabRoby to perform the rehabilitation tasks in a desired manner. Admittance control with inner robust position control loop, which provides the necessary motion to RehabRoby to complete the rehabilitation task in a desired manner, is used. The level of resistance applied by RehabRoby can be varied using admittance control based on the patient's movement capability. Admittance controller has been

integrated with a robust position controller which consists of a linear discrete Kalman filter to compensate effects of the parameter variations, and nonlinearities in the inherent dynamic model of RehabRoby and the external forces that may affect the human–robot interaction. When the disturbances are compensated, it becomes possible to control the position of RehabRoby with feedforward and state feedback techniques using a robust position controller. Furthermore, an admittance control with an inner robust position control loop does not need an exact knowledge of RehabRoby’s dynamic model; thus, the computation effort of the control algorithm is minimized. The high-level controller is the decision-making mechanism that decides the necessary changes in the low-level controller according to the sensory information or the therapist’s commands. A hybrid system modeling technique has been used for the high-level controller, which provides flexibility in interfacing the low-level controller without extensive redesign cost.

RehabRoby can provide passive, active-assisted and resistive-assisted therapy modes; thus, it is possible for low-functioning and high-functioning patients to use RehabRoby in their rehabilitation programs. The transitions between the controllers (when needed) can be completed in a smooth manner without causing any nonlinearities and jerks with the high-level controller, which is an important issue during execution of the rehabilitation tasks.

As future research, the robust position controller of RehabRoby can be improved using an adaptive Kalman filter. The capability of RehabRoby can be extended adding new therapy modes. Additionally, the proposed robot-assisted rehabilitation system RehabRoby will be used in the future for the rehabilitation of stroke patients.

7.7. Acknowledgments

The study was supported by the Support Programme for Scientific and Technological Research Projects (TUBITAK-3501) under Grant 108E190. We gratefully acknowledge the help of Dr. Duygu Geler Külçü, who is in the Department of Physical Medicine and Rehabilitation, Yeditepe University School of Medicine, Yeditepe University Hospital.

7.8. Bibliography

- [CHO 03] CHOI C., KWAK N., “Robust control of robot manipulator by model-based disturbance attenuation”, *IEEE/ASME Transactions on Mechatronics*, vol. 8, no. 4, pp. 511–513, 2003.

- [EMA 04] EMARA H., ELSHAFEI A.L., “Robust robot control enhanced by a hierarchical adaptive fuzzy algorithm”, *Engineering Applications of Artificial Intelligence*, vol. 17, pp. 187–198, 2004.
- [FAS 04] FASOLI S.E., KREBS, H.I., HOGAN N., “Robotic technology and stroke rehabilitation: translating research into practice”, *Topics in Stroke Rehabilitation*, vol. 11, no. 4, pp. 11–19, 2004.
- [FRI 07] FRISOLI A., BORELLI L., MONTAGNER A., *et al.*, “Arm rehabilitation with a robotic exoskeleton in virtual reality”, *Proceedings of the 2007 IEEE 10th International Conference on Rehabilitation Robotics*, Noordwijk, Netherlands, pp. 631–642, 19–23 May 2008.
- [GUL 07] GULEC E., Anthropometric dimensions of human in anatolia, Ankara University Scientific Research Project Report, 2007.
- [HES 05] HESSE S., WERNER C., POHL M., *et al.*, “Computerized arm training improves the motor control of the severely affected arm after stroke”, *Journal of the American Heart Association*, vol. 36, pp. 1960–1966, 2005.
- [HOU 07] HOUSMAN S.J., LE V., RAHMAN T., *et al.*, “Arm-training with T-WREX after chronic stroke: preliminary results of a randomized controlled trial”, *Proceedings of the 2007 IEEE 10th International Conference on Rehabilitation Robotics*, Noordwijk, Netherlands, pp. 562–568, 13–15 June 2007.
- [JUN 98] JUNG S., HSIA T.C., “Neural network impedance force control of robot manipulator”, *IEEE Transactions on Industrial Electronics*, vol. 45, no. 3, pp. 451–461, 1998.
- [KOU 07] KOUSIDOU S., TSAGARAKIS N.G., SMITH C., *et al.*, “Task-oriented biofeedback system for the rehabilitation of the upper limb”, *Proceedings of the 2007 IEEE 10th International Conference on Rehabilitation Robotics*, Noordwijk, Netherlands, 2007, pp. 376–384.
- [KRE 04] KREBS H. I., FERRARO M., BUERGER S.P., *et al.*, “Rehabilitation robotics: pilot trial of a spatial extension for MIT-Manus”, *Journal of NeuroEngineering and Rehabilitation*, vol. 1, pp. 1–15, 2004.
- [KRE 07] KREBS H. I., VOLPE B.T., WILLIAMS D., *et al.*, “Robot-aided neurorehabilitation: a robot for wrist rehabilitation”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 15, no. 3, pp. 327–335, 2007.
- [LIV 65] LIVINGSTONE C., *Joint Motion: Methods of Measuring and Recording*, American Academy of Orthopaedic Surgeons, Edinburg, 1965.
- [LOU 03] LOUREIRO R., AMIRABDOLLAHIAN F., TOPPING M., *et al.*, “Upper limb mediated stroke therapy – GENTLE/s approach”, *Autonomous Robots*, vol. 15, pp. 35–51, 2003.

- [LUM 06] LUM P.S., BURGAR C.G., VAN DER LOOS H.F.M., *et al.*, “MIME robotic device for upper-limb neurorehabilitation in subacute stroke subjects: a follow-up study”, *Journal of Rehabilitation Research & Development*, vol. 43, pp. 631–642, 2006.
- [NEF 07] NEF T., GUIDALI M., RIENER R., “ARMin III-arm therapy exoskeleton with an ergonomic shoulder actuation”, *Applied Bionics and Biomechanics*, vol. 6, no. 2, pp. 127–142, 2009.
- [OLD 07] OLDERWURTEL F., MIHELJ M., NEF T., *et al.*, “Patient-cooperative control strategies for coordinated functional arm movements”, *Proceedings of the European Control Conference*, Kos, Greece, pp. 2527–2534, 2–5 July 2007.
- [OZK 11a] OZKUL F., BARKANA EROL D., DEMIRBAS BADILLI S., *et al.*, “Evaluation of proprioceptive sense of the elbow joint with rehabRoby”, *Proceedings of the IEEE 12th International Conference on Rehabilitation Robotics*, Zurich, Sweden, pp. 1–6, 29 June–1 July 2011.
- [OZK 11b] OZKUL F., BARKANA EROL D., “Design of an admittance control with inner robust position control for a robot-assisted rehabilitation system RehabRoby”, *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2011)*, Budapest, Hungary, pp. 104–109, 3–7 July 2011.
- [OZK 11c] OZKUL F., BARKANA EROL D., “Design and control of an upper limb exoskeleton robot RehabRoby”, *12th Conference Towards Autonomous Robotic Systems*, Lecture Notes in Computer Science, vol. 6856, Sheffield, UK, pp. 125–136, 31 August–2 September 2011.
- [OZK 12] OZKUL F., BARKANA EROL D., DEMIRBAS BADILLI S., *et al.*, “Evaluation of elbow joint proprioception with RehabRoby: a pilot study”, *Acta Orthopaedica et Traumatologica Turcica*, vol. 46, no. 5, pp. 332–338, 2012.
- [ROS 07] ROSATI G., GALLINA P., MASIERO S., “Design, implementation and clinical tests of a wire-based robot for neurorehabilitation”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 15, no. 4, pp. 560–569, 2007.
- [SAL 95] SALVATORE L., STASI S., “LKF based robust control of electrical servodrives”, *IEEE Proceedings Electric Power Applications*, vol. 3, no. 142, pp. 161–168, 1995.
- [SAN 06] SANCHEZ R., LIU J., RAO S., *et al.*, “Automating arm movement training following severe stroke: functional exercises with quantitative feedback in a gravity-reduced environment”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 3, pp. 378–389, 2006.
- [STA 99] STASI S., SALVATORE L., MILELLA F., “Robust tracking control of robot manipulators via LKF-based estimator”, *Proceedings of the IEEE International Symposium*, Bled, Slovenia, pp. 1117–1124, 12 July 1999.

- [TAK 05] TAKAHASHI C.D., DER-YEGHIAIAN L., LE V.H., *et al.*, “A robotic device for hand motor therapy after stroke”, *IEEE 9th International Conference on Rehabilitation Robotics*, CA, pp. 17–20, 28 June–1 July 2005.
- [TSE 07] TSETSERUKOUL D., TADAKUMA R., KAJIMOTO H., *et al.*, “Towards safe human-robot interaction: joint impedance control towards safe human-robot interaction: joint impedance control”, *Proceedings of 16th IEEE International Conference on Robot & Human Interactive Communication*, Jeju, Korea, pp. 860–865, 26–29 August 2007.

Chapter 8

MIMO Actuator Force Control of a Parallel Robot for Ankle Rehabilitation

This chapter presents a multi-input multi-output (MIMO) force controller for a parallel robot that is used for ankle rehabilitation. The robotic device is an advanced rehabilitation system compared to current designs through modeling the patient's ankle as part of the robot in the kinematic design. The MIMO actuator force controller is designed so that the gains along the decoupled directions can be pushed closer to their corresponding gain margins to enhance force-tracking performance over uniform gain approaches. The force controller will be used in the loop to provide an adaptive interaction control to perform repeatable and objective physiotherapy to patients following an ankle injury.

8.1. Introduction

While robots have long been used for automation of industrial processes, there is a growing trend where robotic devices are used to provide services for end users. An area where robots are believed to have a significant impact is healthcare. Accessibility to healthcare services is a vital component to improve the quality of life. However, the trend of aging populations will certainly increase demand on healthcare and create more strain on the already

limited resources available. For this reason, much research has been dedicated to medical and healthcare robots.

8.1.1. *Rehabilitation robots*

Robots have been used for rehabilitation purposes since the 1960s [HAR 95]. The application of robots in rehabilitation was initially more focused on replacing lost functions in individuals with physical disabilities through the use of devices such as robotic orthoses, robotic workstations, feeding devices and robotic wheelchairs [TEJ 00]. Over the last two decades, however, with major advances in mechatronics engineering [HAB 06, HAB 08], there has been an increasing amount of research into the use of robots in physical therapy. One of the main motivations behind the adoption of robots in physical therapy is the potential improvement in productivity [KRE 00]. Physical therapy normally requires manual manipulation of the patient's affected limb, and these manipulations can be rather labor intensive. Consequently, such rehabilitation exercises can easily lead to the onset of fatigue in the therapist, thus limiting the duration and intensity of the therapy session. Since robots are well suited for repetitive tasks and can be designed to have adequate force capabilities, their use in the execution of these exercises will be able to reduce the physical workload of therapists, and can potentially allow the therapists to simultaneously oversee the treatment of multiple patients in a supervisory role [KRE 00, RIE 05].

The use of robots in physical therapy also provides further advantages due to their high repeatability and ability to collect a vast amount of quantitative data when equipped with appropriate sensors. Since therapists mainly operate on the basis of their "feel", their evaluation of the patient's condition can be subjective. By using robotic devices, diagnosis and prognosis can be made more objectively using quantitative data, and comparisons between different cases can also be made more easily [RIE 05]. The high repeatability of the robotic devices also allows therapy to be applied more consistently and will help to identify the effectiveness of the treatment. As a result, in addition to the delivery of physical therapy, robots can also contribute to rehabilitation research.

Existing robots designed for physical therapy are commonly involved with neuromotor training of patients suffering from neurological disorders [HES 03, REI 04]. Robots used in this capacity are generally required to manipulate the patient's affected limb by guiding it along certain motion trajectories. It can be seen that rehabilitation robots share several common

traits. The obvious feature found in all these robots is the emphasis on the user's safety. As the patient is tightly coupled to the rehabilitation robot during its operation, it is vital that the patient-robot interaction forces or torques be maintained at safe levels to prevent any injuries. Therefore, this requires the robotic devices to have some degree of compliance or in other words, to be backdrivable. Inherent backdrivability can be realized by using a low actuator transmission ratio or by decoupling the actuator mass from its end point through the use of elastic elements. Alternatively, force feedback control can be used to reduce the apparent actuator mass and improve the backdrivability of actuators [HOG 87, HOG 05].

Physical characteristics such as size, shape, mass, joint kinematics, motion range and joint dynamics can vary considerably among individuals. Robots designed for rehabilitation must therefore be adjustable or adaptable so that they can cater for a larger population. If these characteristics are not taken into consideration in the robot controller, the robot may become unstable or it may apply excessive forces in non-compliant directions, thus presenting a dangerous scenario for the patient.

Another common feature among rehabilitation robots is the need to control the physical interaction between the patient and the robot. This means that both the motion of the robot and the contact forces applied to the patient must be regulated. Motion regulation is generally required when guiding the patient's limb along paths that are representative of reaching tasks for the upper limb or trajectories which corresponds to normal gait pattern for the lower limb. The requirement to control forces and torques, on the other hand, can arise from concerns of the patient's safety or from the need to apply resistive effort for strength training exercises.

8.1.2. *Ankle sprain rehabilitation*

Ankle sprains are injuries that involve the overstretching of ligaments around the ankle and are often sustained during sporting or physical activities. Due to its location, the human ankle is frequently subjected to large loads, which can reach up to several times the body weight. The exposure to such large loads also means a higher likelihood of injuries. In fact, the ankle is the most common site of sprain injuries in the human body, with over 23,000 cases per day in the United States [HER 02]. In New Zealand, approximately 82,000 new claims related to ankle injuries were made to the Accident Compensation Corporation (ACC) in the year 2000/2001, costing

an estimated 19 million NZD and making ankle-related claims the fourth biggest cost for ACC [ACC 07].

Depending on the severity of the sprain, the time required for recovery can range from 12 days to more than six weeks [SAF 99]. Researchers have reported that a significant number (>40%) of severe ankle sprains can develop into chronic ankle instability [SAF 99], which makes the ankle more susceptible to further injuries in the future. Chronic ankle instability is thought to be caused by a combination of mechanical and functional instabilities at the ankle. Mechanical instability refers to changes of the ankle anatomy that makes it more prone to future sprain injuries, whereas functional instability refers to changes that give rise to insufficiencies in the ankle neuromuscular system, such as impaired proprioception, muscle weakness and reduced neuromuscular control [HER 02].

The general rehabilitation program for ankle sprains is carried out in stages. The initial stage of treatment right after injury is focused on reducing effusion and swelling at the affected area to promote healing of the injured tissues. A reduction in effusion can be achieved with elevation, application of ice and compression. The affected ankle is also often immobilized. However, as prolonged immobilization of the ankle can lead to a reduced range of motion (ROM) and muscular atrophy, the next phase of ankle rehabilitation typically involves ROM and muscle strengthening exercises. Once effusion has stopped, active and passive ROM exercises are normally carried out within the pain-free range of the patient. Research suggests that this has the ability to stimulate healing of torn ligaments [SAF 99]. Once pain-free weightbearing gait is achieved, muscle stretching and resistive exercises can be initiated, where the resistance level should be increased as the patient progresses with recovery. Muscle stretching is important to assist the recovery of joint ROM while resistance training is used to improve the strength of muscles surrounding the ankle to prevent future injuries [MAT 02]. Finally, proprioceptive and balancing exercises should be carried out toward the end of the rehabilitation program to enhance the patients' sense of joint position, thus giving them better foot and ankle coordination and improving their ability to respond to sudden perturbations at the ankle [SAF 99].

As can be seen from the previous discussion, muscular strength and good proprioception are vital in preventing functional instability of the ankle. Therefore, emphasis must be given to these areas and an extensive rehabilitation program is needed to minimize the likelihood of recurrent injuries. The repetitive and tedious nature of such exercises, therefore, makes

robotic devices an attractive alternative to manual manipulation. Various robotic platforms have therefore been developed for the rehabilitation of ankle injuries to reduce the physical workload of therapists and supplement the resources required to facilitate a comprehensive rehabilitation regime so that adequate therapy can be delivered to the patients. However, further development is still required to enhance the adaptability and to improve the suitability of existing devices as clinical measurement tools.

8.2. Ankle rehabilitation robot

Due to the high incidence and potentially lengthy rehabilitation period of ankle injuries, there is a significant demand for the treatment of such injuries. As physical therapy is vital in the promotion of recovery and prevention of future injuries, effort is required to ensure the availability of this service. Introduction of robots in ankle rehabilitation will allow delegation of tedious rehabilitation tasks to the robot, and allows therapists to extend care to more patients. As discussed previously, robots can also be used as evaluation tools to determine the progress and capability of the patients. This means that robots can potentially be used to determine whether a patient has achieved a suitable level of muscle strength and proprioceptive capability required to prevent future injuries.

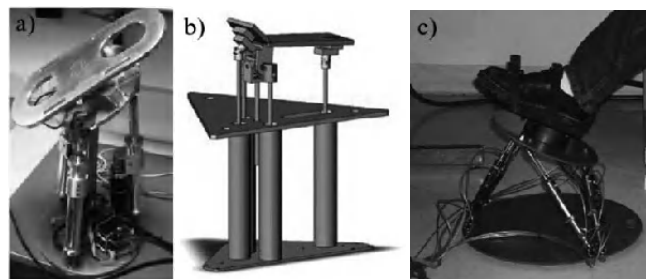


Figure 8.1. Examples of platform based ankle rehabilitation robots. a) The ankle exerciser developed in [BON 07]; b) the reconfigurable ankle rehabilitation robot developed in [BUF 08]; c) the Rutgers ankle rehabilitation interface developed in [YAG 04] (Images reproduced from [ROY 07], [YOO 05] and [TRU 10])

A range of platform-based devices have been developed by researchers for the purpose of sprained ankle rehabilitation [GIR 01, YOO 05, LIU 06, YOO 06, SYR 08, SAG 09], some of which are shown in Figure 8.1. They are designed to carry out various ankle rehabilitation exercises such as

motion therapy and muscle strength training. Motion therapy can be divided into passive, active-assist and active exercises, each requiring a different level of participation from the patient, ranging from no active effort in the passive exercises to full user-driven motion in active exercises. Strength training, on the other hand, requires the robot to apply a resistive load to impede the user's movement to improve muscle strength.

A survey of existing ankle rehabilitation robots shows that the end-effectors of existing platform-based systems are typically constrained about a center of rotation that does not coincide with the actual ankle joint [YOO 05, LIU 06, YOO 06, SAG 10]. A result of this is that the user's shank will not be stationary during operation of these robots if the natural ankle-foot motion is to be maintained. Consequently, orientation and interaction moments of the robotic platform are unlikely to be equivalent to the actual displacement and moments found between the foot and the shank. Therefore, this makes these existing devices less suitable for evaluation purposes due to the greater uncertainties in the motion/force information.

Different actuators were used in platform-based ankle rehabilitation robots. The Stewart platform-based device in [GIR 01] and the reconfigurable ankle rehabilitation platform in [YOO 05, YOO 06] have utilized pneumatic cylinders to provide actuation, while electric motors were used in devices developed in [LIU 06, SUN 07, LIN 08]. A custom-designed electric actuator was proposed in [SAG 09, SAG 10] to improve actuator backdrivability, whereby a cable-driven pulley system is used to convert the rotational motion of a direct current (DC) motor into linear motion of the actuator rod.

8.2.1. Design requirements

To carry out different ankle rehabilitation exercises, the robot to be developed must have a workspace similar to or more than the typical range of motion encountered at the human ankle. The ankle-foot motion is primarily rotational and is often described by rotations on three mutually perpendicular anatomical planes. These rotations are illustrated in Figure 8.2. The plane that distinguishes the left and right sides of the body is called the sagittal plane. The frontal plane, as its name suggests, divides the body into front and back halves. Finally, the transverse plane divides the body into top and bottom portions. Rotational motion of the foot on the sagittal plane is called plantar flexion, which happens when the toes are pushed further away from the head, and rotational motion of the foot in the opposite direction is called dorsiflexion. Inversion is used to describe the rotation of the foot on the

frontal plane where the inner or medial side of the foot is raised upward, with eversion being its complementary motion. Finally, internal rotation or adduction is used to describe the rotational motion on the transverse plane which moves the toes toward the center of the body, whereas the movement in the opposite direction is called external rotation or abduction. The typical motion limits along these different directions as determined in an *in vitro* study by Siegler *et al.* [SIE 88] are shown in Table 8.1.

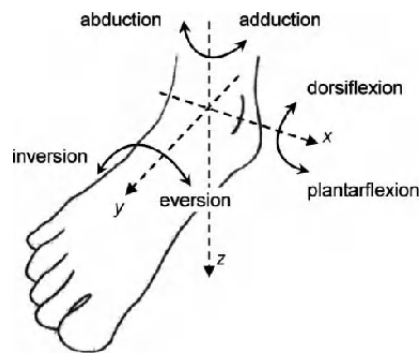


Figure 8.2. Rotational motions of the human ankle

Type of motion	Maximum allowable motion		
	Range (deg)	Mean (deg)	Standard deviation (deg)
Dorsiflexion	20.3–29.8	24.68	3.25
Plantar flexion	37.6–45.75	40.92	4.32
Inversion	14.5–22	16.29	3.88
Eversion	10–17	15.87	4.45
Internal rotation	22–36	29.83	7.56
External rotation	15.4–25.9	22.03	5.99

Table 8.1. Typical range of motion at the human ankle [SIE 88]

It can be seen that the extents of motion available in different directions are quite different and the overall ankle range of motion is small. It should be noted that since the robot should be able to cater for both the left and right legs, the different motion limits in the inversion–eversion and internal–external rotation directions will be inverted in the robot coordinate frame

when a foot from the different side of the body is placed on the robot. The limits of the required robot rotational workspace on the frontal and transverse planes are therefore symmetric.

Another quantity that has a significant influence on the design of the ankle rehabilitation robot is the level of moment that the ankle-foot structure is expected to experience during rehabilitation. In terms of the maximum moment required at the plantar-/dorsiflexion motion, results from an *in vivo* study in [KEA 90] confirm that a maximum range of 71.7 Nm is required to move the foot of the subject passively from the maximum plantar flexion to the maximum dorsiflexion. The same study also evaluated the torques produced by maximum voluntary contraction of the subjects and the corresponding values for dorsiflexion and plantar flexion are 54.4 Nm and 126.0 Nm, respectively. Similar results in terms of passive ankle moments were also observed in an *in vitro* study by Paranteau *et al.* [PAR 98], which gives a maximum dorsiflexion moment as -44 Nm and a maximum plantar flexion moment of about 37 Nm. Maximum joint torque in the inversion-eversion directions is also available from [PAR 98], where values of 33 Nm in inversion and 44 Nm in eversion were reported. Unfortunately, maximum torque for internal/external rotation is not available from the above studies. The robot used in this research was designed by assuming that the maximum internal/external rotation moments are similar in magnitude to the inversion/eversion moments. In summary, the moment requirements of the ankle rehabilitation robot are set at 100 Nm for moments about the X Euler angle axis, and 40 Nm for the remaining two Euler angles axes.

In terms of functionality, ankle rehabilitation robots will have to be able to accommodate different types of rehabilitation exercises, which include passive, active-assist and active ROM exercises, as well as muscle strengthening routines. Passive ROM exercises will involve the robot guiding the user's foot through its permissible range of motion when the user's foot remains relaxed. Active-assist ROM exercises, on the other hand, require the robot to "cooperate" with the user to perform the required foot motion, providing assistance on an as-needed basis, while active ROM exercises provide full control of the foot motion to the user, with the robot providing minimal interaction forces/moments. As for the realization of muscle strength training exercises, the robot should be able to provide a constant level of resistance to the foot or vary the resistance according to the extent of displacement (i.e. act as an elastic element).

8.2.2. Analysis for four-link parallel mechanism

Parallel mechanisms have a kinematic structure whereby the end-effector is connected to a fixed base through multiple actuated links. Due to this arrangement, parallel robots have several advantages over their serial counterparts. One of these advantages is a higher positioning accuracy since errors in the actuated joints no longer accumulate as in the case of serial robots. Furthermore, since the end-effector is supported by multiple actuators, the load capacity of the mechanism can also be increased. As actuators of a parallel robot are located at its base rather than on its moving links, the total load moved by the manipulator is also reduced. As a result, parallel mechanisms can be used to achieve higher bandwidth in motion (e.g. Delta robot).

Due to its many advantages and the relatively large loads experienced at the ankle and foot, parallel mechanisms are excellent candidates for ankle rehabilitation devices. In fact, the human lower leg and foot can itself be viewed as a parallel mechanism with the foot being the end-effector and the muscles spanning across the ankle being the actuating links. From the above discussion, it can be seen that there is sufficient motivation for the use of a parallel robot in this research. The major shortcomings of parallel mechanisms, however, come in the form of a reduced workspace and increased kinematic singularities [TSA 99, MER 06]. Fortunately, as the ROM of the human ankle is limited, the smaller workspace of parallel manipulators may still be adequate, provided that suitable kinematic parameters are selected for the mechanism. However, singularities pose a much greater concern and must be considered in the design of the manipulator.

Due to the incorporation of the human ankle as part of the parallel mechanism, its kinematic description must be established before any analyses on the workspace, singularities and moment capabilities of the ankle rehabilitation robot. Although foot motion is often depicted through rotations about two oblique revolute joints in series [INM 76, DUL 85, SIM 68], its actual movement pattern appears to be more complicated with coupled translations and rotations. Studies found that the orientations of the revolute joints in the biaxial model can vary significantly among individuals. Furthermore, it was also found that such axes orientations also vary with the configuration of the foot. Based on these findings, the generality of results obtained from using a specific biaxial ankle model in the workspace and singularity analysis would be compromised. A natural choice of a kinematic model to replace the biaxial model is a spherical joint as it can cater for all

possible rotational motions. However, this approach still fails to address the effects of translational motion. As the movement of the ankle can be considered primarily rotational with limited translational movement of its instantaneous axis of rotation [YIN 05], analyses that consider the ankle as a spherical joint can still be used to give an indication of the available workspace and singular regions.

After developing a conventional parallel robot mathematical formulation, as per [MIT 03], and carrying out a full design analysis, including the workspace, singular region and actuator force analyses that is presented in [TSO 10], a four-link parallel mechanism was chosen by the authors. The attachment points of the four-link mechanism on the end-effector and the base platform considered in this research share the same parameterization as the three link version. The kinematic structure of the mechanism can therefore be represented by Figure 8.3.

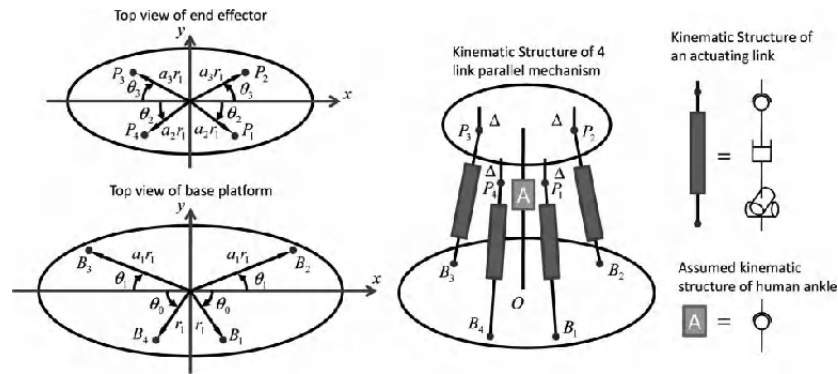


Figure 8.3. Kinematic structure of the three-link parallel mechanism

In Figure 8.3, the attachment points of the actuated links on the base are denoted by B_i while their attachments on the end-effector are represented by P_i . Based on the UPS link structure, point B_i is coincident with the center of the universal joint while point P_i is coincident with the center of the spherical joint or equivalent on the i th actuated link. Point O had also been defined on the base platform where it acts as the origin of the robot global coordinate frame. The points B_i and O are constrained to lie on the same plane and their relative positions are parameterized in polar coordinates. The projections of points P_i on the end-effector can similarly be represented in polar coordinates. In addition, the distance between P_i and the end-effector plane is

also set to be constant for all i , and is denoted by Δ . Finally, the point A is defined as the center of the spherical joint used to represent the human ankle.

Using the proposed kinematic structure, the end-effector can be seen to pivot about the actual ankle center and not an external point. Consequently, when the shank is fixed on the base platform and the foot is placed on the robot end-effector, the robot would have completely isolated the ankle joint. Motion and moments of the end-effector taken about the ankle center will therefore, respectively, provide accurate indications of the relative orientation and moments between the user's foot and shank.

The workspace and condition numbers for the final design were also computed for different locations of the spherical joint (i.e. OA) to simulate the translation of the actual ankle center of rotation. In the final design analysis, OA was varied within a 30 mm cube centered about its nominal location. However, positions of the end-effector attachment points, $P_{i,o}$, were held constant with respect to the global robot frame. The choice of the 30 mm cube was based on sources in the literature which estimates the range of ankle translation to be between 10 and 20 mm [YIN 05]. It should be noted that the workspace produced from this analysis will also have some safety margin against small deviations in the foot placement location on the robot end-effector. The upper and lower bounds of the actuator lengths used in this analysis also differed slightly with those of the original analysis where actuator length limits are dependent on the actuator lengths at the neutral foot configuration. In the final design, the construction of the actuated links was made to be uniform across all links and they therefore share the same motion limits. Analysis of the reachable workspace had shown that this design decision had the effect of improving the maximum motion allowable in the plantar flexion direction at the expense of smaller dorsiflexion movements. Since the original dorsiflexion motion limit is much more than the natural dorsiflexion motion limit, this design change is not expected to significantly compromise the ability of the proposed robot in meeting the workspace requirements.

The workspace, singular region and actuator force analyses carried out on the four-link mechanism with kinematic parameters are given in Table 8.2 and the results for a slice of the robot workspace are shown in Figure 8.4. The kinematic parameters of this parallel mechanism also appear to be capable of producing a larger robot workspace in the flexion directions, with maximum plantar flexion of about 52° and maximum dorsiflexion of about 48° . The motion limits about the Y-Euler axis, however, was found to

decrease slightly to 34° , but it still satisfies the required range of motion. Finally, the range of Z-Euler rotations in the workspace is also more than adequate to accommodate the natural ankle movements. An inspection of the actuator force requirements also shows that with four actuators, the maximum actuator force exerted to achieve the prescribed task space moment is now reduced to about 1,700 N from over 3,000 N in the three-link mechanism.

Parameter	r_1	a_1	a_2	a_3	θ_0	θ_1	θ_2	θ_3	Δ	$\overline{OA}$
Value	0.2 m	0.9	0.4	0.45	-45°	45°	-30°	30°	0.05 m	$[0 \ 0 \ 0.36]$ m

Table 8.2. Kinematic parameters for the four-link parallel mechanism

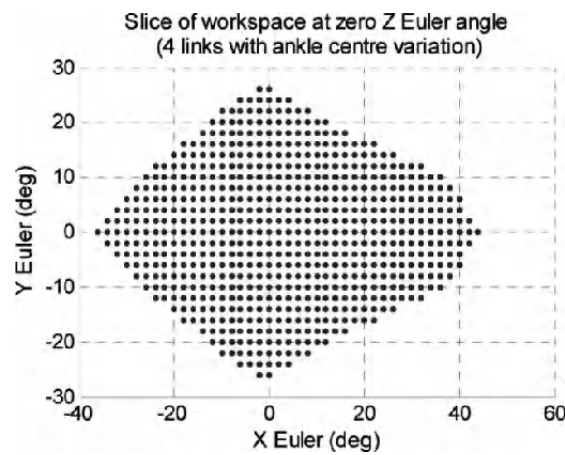


Figure 8.4. The common robot workspace at zero Z-Euler angle. The information shown is for the four-link parallel mechanisms obtained by varying $\overline{OA}$ within a 30 mm cube centered about its nominal value

The results obtained from the above analysis show that the estimated range of motion can be considered adequate, with a maximum plantar flexion of 44° , maximum dorsiflexion of 36° and maximum inversion–eversion of 26° . The available abduction and adduction motion again remains large at about 70° . The moment capacity of the final design can also be evaluated by considering the maximum forces available from the actuators. This can be done by considering the maximum moments that can be applied at all

end-effector orientations which belong to the common workspace and taking the smallest of these values. Note that this moment analysis was carried out on each of the X, Y and Z directions by using a maximum actuator force output of 2,000 N. The moment capacity and the maximum achievable end-effector orientations of the final design are summarized in Table 8.3. It can be seen that the movements and moments achievable by the four-link parallel mechanism are similar to what is required for the X and Y directions, and more than what is needed in the Z direction, thus indicating the capability of the proposed structure to perform the required rehabilitation exercises.

Direction	Maximum motion	Moment capacity
Plantar flexion (positive X)	44°	151 Nm
Dorsiflexion (negative X)	36°	151 Nm
Inversion (positive Y)	26°	38 Nm
Eversion (negative Y)	26°	38 Nm
External rotation/Abduction (positive Z)	72°	68 Nm
Internal rotation/Adduction (negative Z)	72°	68 Nm

Table 8.3. *Motion limits and moment capacity of the four-link parallel mechanism*

8.2.3. System description

A prototype of the ankle rehabilitation robot had been constructed using the kinematic parameters investigated above. Brushed DC motor-driven linear actuators (Ultra Motion Bug Linear Actuator 5-B.125-DC92_24-4-P-RC4/4) had been used as the actuated prismatic joint in the prototype. The linear actuator was chosen based on the stroke length and force requirements of the mechanism, with an actuator stroke length of 0.1 m, force capacity of over 2,000 N and a top speed of 0.066 m/s. In terms of sensors, linear potentiometers were built into the actuators to provide measurement of the actuator lengths. Additionally, a two-axis inclinometer (Signal Quest SQ-SI2X-360DA) was attached to the end-effector platform to allow the measurement of its pitch and roll angles. Lastly, four tension–compression load cells (Omega Engineering LC201-300) were also installed at the interface between the linear actuator and the effective spherical joints to monitor the forces along the actuated links. The ankle rehabilitation robot developed in this research is shown in Figure 8.5, both in the form of a three-dimensional model and in the form of a photograph depicting how the robot interacts with the user. In terms of controller hardware, an embedded

controller (National Instruments NI-PXI 8106) had been used together with a Data Acquisition (DAQ) card (National Instruments NI-PXI-6229) to carry out the signal processing and execute the real-time control functions of the prototype. The embedded controller was also connected to a *personal computer* to receive user commands and display the sensor measurements through a user interface developed using the LabView programming environment.

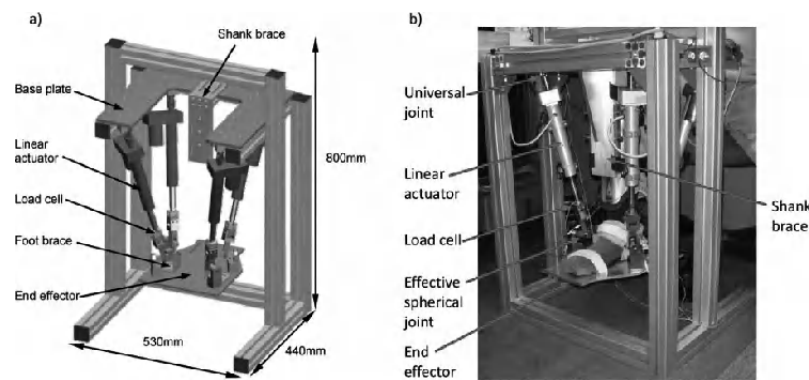


Figure 8.5. The 3D CAD model of the developed ankle rehabilitation robot a) and a photograph showing the robot with the user's lower limb attached b)

8.3. Actuator force control

Operation of the rehabilitation robot relies on implementation of a suitable interaction controller, and a force-based control approach has been taken, whereby the desired robot impedance is realized through a MIMO actuator-level force control. Preliminary experimental trials carried out in this research revealed that independent application of actuator force control schemes (as are conventionally implemented) on each actuator yielded undesired oscillatory behavior. Consequently, the MIMO actuator force controller that takes into account the coupling effects introduced by kinematic constraints of the parallel manipulator was used to improve the force control performance.

The design of a force controlled loop has been considered in this research and although various actuator force/torque control techniques are available in the literature [MUR 93, KAT 06, KAT 08, KON 08], they are mainly focused on the control of a single actuator as a single-input single-output (SISO) system, and do not consider the overall kinematics of the mechanism being

driven. These schemes also often require measurements of velocities and good knowledge of the robot dynamics. As significant force coupling can be found among the actuator forces in a parallel robot, independent actuator force control may lead to performance degradation as force interactions between various actuators will simply be treated as additional disturbances in the SISO force controllers. A MIMO approach for actuator force control that takes into account the kinematic and inertial information of the parallel manipulator is presented here.

8.3.1. Actuator force control by decoupling of inertia matrix

The dynamics of the actuators used in this research has to be considered to gain a better understanding of the coupling that arises when these actuators are installed on the parallel robot. The equation governing the dynamics of the brushed DC motor-driven linear actuators used in this research is given here [8.1], where $\mathbf{F}_{act}$ is the actuator force resisting the actuator motion, i is the actuator current, l is the actuator length, K_t is the torque constant of the motor, K_a is the transmission ratio between the rotary and linear displacements, J_{eff} is the effective motor rotational inertia, b_{eff} is the effective motor viscous damping coefficient and F_{fric} is the effective Coulomb friction acting on the actuator rod.

$$F_{act} = K_t K_a i - K_a^2 J_{eff} \ddot{l} - K_a^2 b_{eff} \dot{l} - F_{fric} \quad [8.1]$$

Provided that the actuators share the same characteristics, equation [8.1] will be equally applicable for the case where the actuators are coupled through a parallel mechanism. In such a scenario, the variables F_{act} , i , l and F_{fric} can simply be treated as vectors. This relationship can, therefore, be integrated with the task space dynamics to give the overall actuator and robot dynamics. As with many manipulators, the robot task space dynamics considered in this chapter can be represented by equation [8.2], where Θ is the task space coordinates in XYZ Euler angles, $M(\Theta)$ is a configuration-dependent inertia matrix, $N(\Theta, \dot{\Theta})$ is a vector gathering all the centripetal, Coriolis and gravitational forces, τ_{ext} is the external torque applied to the robot manipulator and J is the manipulator Jacobian. Using this manipulator Jacobian, the task space and joint space velocities and accelerations can also be related through equations [8.3] and [8.4], respectively.

$$M(\ddot{\Theta}) + N(\Theta, \dot{\Theta}) + \tau_{ext} = J^T F_{act} \quad [8.2]$$

$$\dot{i} = J\dot{\Theta} \quad [8.3]$$

$$\ddot{i} = J\ddot{\Theta} + \dot{J}\dot{\Theta} \quad [8.4]$$

The actuator accelerations can be obtained by substituting the task space accelerations obtained from the rearrangement of equation [8.2] into equation [8.4]. The resulting actuator accelerations and the actuator velocities can then be further incorporated into equation [8.1] to give equation [8.5], where the matrix D is used to describe the coupling between the actuator currents and the resulting actuator forces. The expression for this matrix is given in equation [8.6]. It should be noted that dependencies on the task space coordinates and velocities are dropped hereafter for brevity. Additionally, since the manipulator Jacobian is not square for redundantly actuated robots, the pseudo-inverse (denoted by the operator $+$) has been used in place of the conventional matrix inverse operation. More specifically, the pseudo-inverse of the manipulator Jacobian is computed using equation [8.7]. It is also worth noting that $J^T(J^+)^T = I$ and $(J^+)^T J^T = I - V_0 V_0^T$, where v_0 is a column-wise collection of the null space vector(s) of J^T .

$$F_{act} = K_i K_a D i + (I - D)(J^+)^T (N + \tau_{ext}) - D \underbrace{[K_a^2 (J_{eff} J + b_{eff} J) \dot{\Theta} - F_{fric}]}_{F_{distb}} \quad [8.5]$$

$$D = \left(I + K_a^2 J_{eff} J M^{-1} J^T \right)^{-1} \quad [8.6]$$

$$J^+ = (J^T J)^{-1} J^T \quad [8.7]$$

Disturbance observer-based controllers have been used by several researchers for position and force control of electric actuators [MUR 93, KEM 99, WAN 04, KAT 06, KAT 08]. Due to its relative simplicity and good performance, this control structure was used in this chapter as the basis for further analysis and development. By treating the last two terms in equation [8.5] as disturbances, the plant that relates the actuator current to the actuator force can be viewed as a simple gain matrix. The actuator force control problem can therefore be represented using the block diagram as shown in Figure 8.6(a). By applying the typical disturbance observer-based control scheme to this system, the resulting closed-loop system is given in Figure 8.6(b). It can be seen in Figure 8.6(b) that the feedback block used in this scenario need only be a gain matrix instead of a low-pass filter as often used in other applications. This is the result of allocating the other dynamical terms as system disturbances to be rejected. Using the proposed controller,

the commanded actuator current can be written as equation [8.8], with K_f being the controller gain used in the feedback path of the disturbance observer, F_c being the vector of commanded force, G being the gain matrix in the forward path and H being the gain matrix in the feedback path. This gain K_f can take on values between zero and 1, and better disturbance rejection can be achieved as the gain approaches unity. The proposed control law can be seen as a simpler version of other similar disturbance observer-based approaches found in the literature as it does not require measurement of actuator velocity or acceleration. This is because the robot inertia matrix and manipulator Jacobians are purely dependent on the configuration of the manipulator end-effector.

$$i_c = \frac{1}{K_t K_a (1 - K_f)} D^{-1} F_c - \frac{K_f}{K_t K_a (1 - K_f)} D^{-1} F_{act} = \frac{1}{K_t K_a} (G F_c - H F_{act}) \quad [8.8]$$

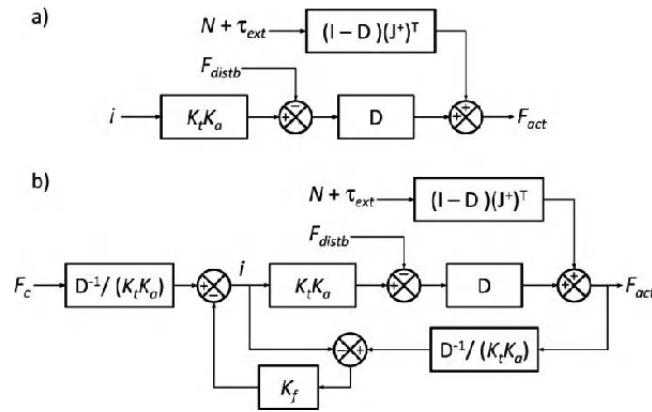


Figure 8.6. Block diagrams of the actuator dynamics a) and block diagram of the system under disturbance observer-based control b).

8.3.1.1. Benefits of decoupling

Substitution of the control law equation [8.8] into equation [8.5] then yields equation [8.9], which shows no coupling between the commanded force and the actual actuator force. An additional benefit of the application of the decoupling gain matrix can be seen by considering the effect that the control law equation [8.8] has on task space accelerations. One of the fundamental uses of actuator force control is to provide improved backdrivability to the rehabilitation robot. This is equivalent to having a zero vector as the commanded force. Substituting equation [8.8] with zero

commanded force into the original actuator force dynamics equation [8.1] then yields equation [8.10], with F_{distb} being a disturbance force which includes frictional and velocity-dependent components of the actuator dynamics. Combination of equations [8.10] and [8.2] will further lead to equation [8.11], which shows that the effective inertia matrix of the mechanism is dependent on the feedback gain matrix H . It can be shown that the effective inertia matrix in equation [8.11] can be eventually simplified to the form shown in equation [8.12] if H is a multiple of the decoupling matrix D^{-1} , with $m_a = K_a^2 J_{eff}$ and $h = K_f / (1 - K_f)$.

$$F_{act} F_c + (1 + K_f)(I - D)(J^+)^T (N + \tau_{ext}) - (1 - K_f) D F_{distb} \quad [8.9]$$

$$F_{act} = (I + H)^{-1} [-K_a^2 J_{eff} J \ddot{\Theta} - F_{distb}] \quad [8.10]$$

$$[M + K_a^2 J_{eff} J^T (I + H)^{-1} J] \ddot{\Theta} + N + \tau_{ext} = -J^T (I + H)^{-1} F_{distb} \quad [8.11]$$

$$M + m_a J^T (I + H)^{-1} J = M + \left[\frac{(1 + h)}{m_a} V_j \sum_J^{-2} V_J^T + h M^{-1} \right]^{-1} \quad [8.12]$$

Closer inspection of equation [8.12] reveals that the second term on the right-hand side is mainly dominated by the term hM^{-1} if m_a is large, which means that the effective inertia matrix would be very similar to a scaled version of the original manipulator inertia matrix. As a result, provided that the robot inertia matrix is diagonally dominant and has a relatively small condition number (this was found to be the case with the developed robot), externally applied torques will mainly contribute to accelerations along similar directions. Since the externally applied torque represents the interaction between the user and the robot, the above implies that the use of the decoupling matrix in the feedback path will allow more intuitive motion of the robot when the user applies an effort to backdrive the manipulator. On the other hand, if an identity matrix is used in the feedback path, the resulting effective inertia matrix of the controlled manipulator will be heavily influenced by the manipulator Jacobian, in particular when the feedback gain is small. In fact, the condition number of the second term in the effective inertia matrix will be the square of that for the manipulator Jacobian, indicating that coupling will be amplified in general even with relatively well-conditioned manipulator Jacobians.

8.3.1.2. Generalization of the decoupling force controller

The control law in equation [8.8] can be further generalized to allow a variation in the level of “decoupling” and the application of additional control action in the null space of the manipulator Jacobian transpose. This modified control law is given in equation [8.13], where n is the power to which the matrix D^{-1} is raised ($0 \leq n \leq 1$) and K_{v0} is the gain for the additional disturbance observer applied along the null vector of the manipulator Jacobian. The main rationale for the addition of a disturbance observer loop in the null space is as follows. Since forces along the null space do not influence the task space moments and motion, its control parameters can be chosen independently. Furthermore, in this particular application, the singular value of the matrix D^{-1} along the null vector is significantly lower than that of the other directions due to the large actuator transmission ratio. The result of this is a smaller control action and a corresponding decrease in disturbance rejection capability along the null space. The addition of the disturbance observer in the null space, therefore, aims to compensate for the above shortcoming.

$$i_c = \frac{1}{K_f K_a} \left[\left(\frac{1}{(1-K_f)} D^{-n} + \frac{1}{(1-K_{v0})} v_0 v_0^T \right) F_c - \left(\frac{K_f}{(1-K_f)} D^{-n} + \frac{K_{v0}}{(1-K_{v0})} v_0 v_0^T \right) F_{act} \right] \quad [8.13]$$

The main motivation for the introduction of partial decoupling through variation of the negative power applied to the D matrix is discussed below. Due to the presence of unmodeled dynamics, there exists a limit for K_f above which the system will become unstable. The introduction of partial decoupling, therefore, allows the controller to be further fine-tuned so that the relative gains applied along each decoupled direction can be changed to strike a balance between the disturbance rejection capability and extent of decoupling while maintaining the overall system stability. The above can be illustrated more clearly by considering a simplified problem. By making the assumption that the force control problem can be perfectly decoupled into four single degree of freedom systems (each lying in one of the decoupled directions or output basis vectors of the D matrix), it can be seen that each of these degrees of freedom will have its own gain margin or critical gain. Clearly, the ratios of these gain margins may not be identical to those of the singular values obtained from the decoupling matrix D^{-1} . It is easy to see that a stable system can only be achieved if the effective gains along all the decoupled directions are less than their corresponding critical gains. Since the effective gains along each of these decoupled directions are dependent on both the singular value of D^{-1} along that direction and the overall controller

gain K_f , the restriction of the relative ratios of singular values in H to that found in D^{-1} will generally lead to the case where effective gains along the decoupled directions are not maximized. A possible approach to alleviate this problem is through the introduction of partial decoupling, which provides an additional controller parameter that can be adjusted so that the effective gains can be pushed closer to their critical values while also maintaining some level of decoupling.

8.3.2. Higher order dynamic model of actuator–sensor–environment system

Stability of force feedback controllers are often compromised due to the presence of unmodeled dynamics and non-collocation of sensor and actuator [AN 87]. The main implication of this phenomenon is the existence of an upper force feedback gain limit. The higher order dynamics introduced by compliance of the force sensor and actuator must, therefore, be considered for a better understanding of the stability of a force feedback controlled system. Higher order dynamics in the force sensor-based actuator force control problem was modeled in [BUE 06, BUE 07] as a three-mass system with two masses representing the actuator and one mass representing the force sensor. This had been done to describe both the first resonance mode of the actuator and the compliance introduced by the force sensor. By assuming a rigid interface between the force sensor and the interacting mechanical environment, the environmental dynamics can also be integrated into the three-mass model to allow for an analysis on the overall system stability. In this chapter, this combined system is referred to as the actuator–sensor–environment system. It should also be noted that when viewed from the perspective of the actuators, the end-effector dynamics of the parallel manipulator as well as any associated kinematic coupling are considered as part of the environment.

8.3.2.1. Three-mass model of the actuator and force sensor hardware

By using a generic three-mass model, the higher order actuator dynamics are found by fitting the actual system into the general model structure. The actual actuator and force sensor hardware used in this research can be modeled as a three-mass system by considering the actuator compliance, sensor compliance and environmental dynamics. Formation of this three-mass model, therefore, requires a more detailed look into the actuator hardware to identify the source of its compliance.

The linear actuators used in this research are powered by brushed DC motors, where the motor torque is transmitted to a ball screw via a 5:1 belt drive. The rotational motion is, in turn, converted into a linear motion through the ball screw with a 3.175 mm (1/8") pitch. Taking into account the compliance of the belt drive, the vibration mode between the motor and the ball screw is represented graphically in Figure 8.7, where the variables θ , ω , J , r are used to denote angular displacements, angular velocities, rotational inertias and pulley radii respectively. The above variables are also used with subscripts m and s to, respectively, represent quantities relating to the motor and the ball screw. Additionally, k_b and b_b are the stiffness and damping of the transmission belt while b_m and b_s are damping introduced by viscous friction on the motor and ball screw. Finally, K_t , i and τ_{load} are, respectively, used to refer to the motor torque constant, motor current and the load torque applied at the ball screw. Taking the Laplace transform of the motor and ball screw dynamics into account then leads to equations [8.14] and [8.15].

$$[J_m s^2 + (b_m + 2r_m^2 b_b)s + 2r_m^2 k_b]\theta_m(s) = K_t I(s) + 2(r_m r_s b_b s + r_m r_s k_b)\theta_s(s) \quad [8.14]$$

$$[J_s s^2 + (b_s + 2r_s^2 b_b)s + 2r_s^2 k_b]\theta_s(s) = -\tau_{load}(s) + 2(r_m r_s b_b s + r_m r_s k_b)\theta_m(s) \quad [8.15]$$

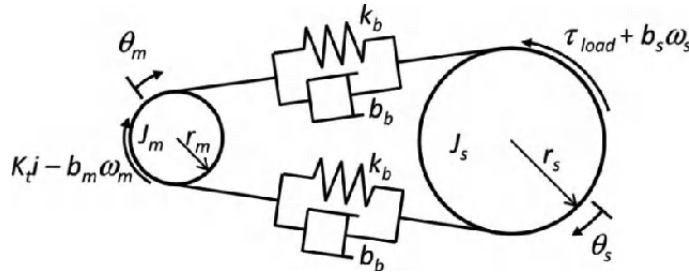


Figure 8.7. Loading conditions between the motor rotor and the ball screw

Assuming a perfectly rigid transmission between the ball screw and the linear actuator rod, the belt drive compliance would become the sole source of actuator compliance, thus resulting in a two-mass model for the actuator. Addition of the force sensor onto the end of the actuator rod then results in the setup shown in Figure 8.8, where x_r and x_f are the displacements of the actuator rod and force sensor, $G_s = 2\pi/\text{pitch}$ is the transmission ratio of the ball screw, m_r is the mass of the actuator rod, b_r is the viscous friction acting on the actuator rod and F_{int} is the interaction force between the force sensor

and the environment. Also, m_f , b_f and k_f are the mass, damping and stiffness of the force sensor, respectively. Analysis of the dynamics of this system leads to equations [8.16] and [8.17]. It should be noted that for completeness, a force F_{fric} is also applied onto the rod to represent Coulomb friction. However, this force is neglected in further analysis within this section as the relationship of interest is that between the input current and measured force.

$$[m_r s^2 + (b_r + b_f)s + k_f]X_r(s) = G_s \tau_{load}(s) + (b_f s + k_f)X_f(s) - F_{fric}(s) \quad [8.16]$$

$$[m_f s^2 + b_f s + k_f]X_f(s) = -F_{int}(s) + (b_f s + k_f)X_r(s) \quad [8.17]$$

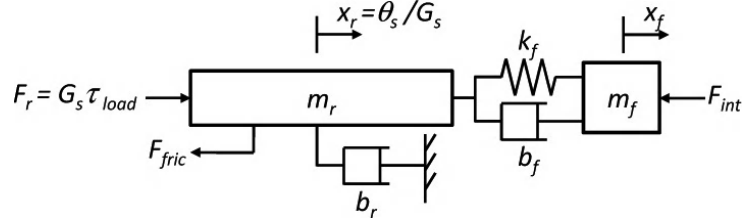


Figure 8.8. Loading conditions between the actuator rod and the force sensor

To complete the definition of the actuator–sensor–environment system, the interaction force shown in Figure 8.8 needs to be related to the dynamics of the interacting environment. In reality, this environment would involve the kinematic coupling imposed by the parallel mechanism. However, for the purpose of illustrating the relationship between the actual actuator parameters and the system parameters of the three-mass model given in the previous section, the force sensor is considered to be rigidly coupled to a single degree of freedom environment through a certain transmission ratio as shown in Figure 8.9. The reason for the introduction of this transmission ratio σ is to allow incorporation of the manipulator Jacobian singular values into the unidirectional model. This arrangement basically considers the unidirectional system to be acting along one of the output basis vectors of the manipulator Jacobian, with the transmission ratio being a factor which scales the task space motion to joint space motion. As the motion variable shown in Figure 8.9 is that of the task space displacement, the associated environmental parameters are also given in the task space. The Laplace transform of the environmental dynamics can be written as equation [8.18].

$$\frac{1}{\sigma^2} [m_e s^2 + b_e s + k_e] X_f(s) = F_{\text{int}}(s) \quad [8.18]$$

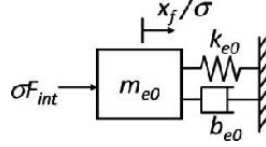


Figure 8.9. Environment dynamics along one of the output basis vectors of the manipulator Jacobian

By considering all the dynamic equations above and using the parameterization shown in equation [8.19], the actuator–sensor–environment system can be restated as the three equations shown in equations [8.20]–[8.22].

$$\frac{r_s}{r_m} G_s X_m(s) = \theta_m(s) \quad [8.19]$$

$$\left[G_s^2 \frac{r_s^2}{r_m^2} J_m s^2 + G_s^2 \left(\frac{r_s^2}{r_m^2} b_m + 2r_s^2 b_b \right) s + 2G_s^2 r_s^2 k_b \right] X_m(s) = \frac{r_s}{r_m} K_t G_s I(s) + 2G_s^2 (r_s^2 b_b s + r_s^2 k_b) X_r(s) \Rightarrow d_m(s) X_m(s) = \frac{r_s}{r_m} K_t G_s I(s) + n_{rm}(s) X_r(s) \quad [8.20]$$

$$\begin{aligned} & [(m_r + G_s^2 J_s) s^2 + (b_r + b_f + b_s G_s^2 + 2G_s^2 r_s^2 b_b) s + k_f + 2G_s^2 r_s^2 k_b] X_r(s) \\ &= 2G_s^2 (r_s^2 b_a s + r_s^2 k_a) X_m(s) + (b_f s + k_f) X_f(s) \Rightarrow d_r(s) X_r(s) \\ &= n_{mr}(s) X_m(s) + n_{fr}(s) X_f(s) \end{aligned} \quad [8.21]$$

$$\left[\left(m_f + \frac{m_{e0}}{\sigma^2} \right) s^2 + \left(b_f + \frac{b_{e0}}{\sigma^2} \right) s + k_f + \frac{k_{e0}}{\sigma^2} \right] X_f(s) = (b_f s + k_f) X_f(s) \quad [8.22]$$

8.3.2.2. Inclusion of kinematic coupling

The model of the actuator–sensor–environment system discussed thus far has been restricted to motion along a unidirectional path. The extension of this model to one that can be representative of the actuator force control on the developed parallel mechanism, therefore, requires the incorporation of the manipulator kinematics and dynamics as part of the mechanical environment.

To simplify this task, the manipulator can be linearized about a particular operating point so that the manipulator Jacobian matrix J can be used to relate the task space and joint space motion variables as shown in equation [8.23], where Θ is the task space coordinates and X_f is a vector of force sensor displacements for all the actuating links. The combined dynamics of the manipulator and the task space mechanical environment can also be approximated in a linear form as in equation [8.24], where M_e , B_e and K_e are the inertia, damping and stiffness matrices of the manipulator and task space environment.

$$X_f(s) = J\Theta(s) \quad [8.23]$$

$$(M_e s^2 + B_e s + K_e)\Theta(s) = J^T F_{\text{int}}(s) \quad [8.24]$$

Since the matrix J is rectangular with more rows than columns due to the use of a redundantly actuated manipulator, by substituting $\Theta(s) = J^+ X_f(s)$ into equation [8.24] and pre-multiplying the resulting equation by $(J^+)^T$, equation [8.25] can be obtained. As before, v_0 is used to denote the null space of J^T and superscript $+$ is used to represent the pseudo-inverse operation.

$$\underbrace{[(J^+)^T M_e J^+ s^2]}_{M'_e} + \underbrace{[(J^+)^T B_e J^+ s]}_{B'_e} + \underbrace{[(J^+)^T K_e J^+]}_{K'_e} X_f(s) = (I - v_0 v_0^T) F_{\text{int}}(s) \quad [8.25]$$

To obtain the overall dynamics for all four actuators, equations [8.20] and [8.21] can still be reused with variables X_M , X_r and X_f now being vectors. However, the dynamic equations for the force sensor masses have to be re-evaluated due to the coupling which exists between the task space and joint space motion variables. The force sensor dynamics before consideration of the interaction force exerted by the environment is given by equation [8.26]. Due to the redundant actuation, the joint space motion along the null space of J^T is constrained to be zero (i.e. $v_0^T X_f(s) = 0$). This fact can then be used to produce equation [8.27]. The combination of equations [8.25], [8.26] and [8.27] then results in equation [8.28]. This equation, together with equations [8.20] and [8.21], can be used to fully describe the dynamics of the coupled actuator–sensor–environment system.

$$\begin{aligned} (m_f s^2 + b_f s + k_f) X_f(s) &= -F_{\text{int}}(s) + (b_f s + k_f) X_r(s) \\ \Rightarrow d_f(s) X_f(s) &= -F_{\text{int}}(s) + n_{rf}(s) X_r(s) \end{aligned} \quad [8.26]$$

$$v_0 v_0^T F_{\text{int}}(s) = n_{rf}(s) v_0 v_0^T X_r(s) \quad [8.27]$$

$$[(M_e' s^2 + B_2' s + K_e') + d_f(s)I] X_f(s) = n_{rf}(s) (I - v_0 v_0^T) X_r(s) \quad [8.28]$$

8.3.3. State space model of the linearized actuator–sensor–environment system

A more general approach in analyzing the stability of the overall actuator–sensor–environment system under closed-loop control would be to analyze the roots of its characteristic equation. This can be a difficult task if approached from the consideration of the coupled transfer function matrix. However, when the system is expressed as differential equations in a time domain and formulated as a state space model, solution of the system poles simply becomes an eigenvalue problem for the state transition matrix.

The state space model for the coupled system can be easily formulated from the kinematic and dynamic relationships of the motor, ball screw, force sensor and the parallel manipulator as presented in equations [8.14]–[8.17], [8.23] and [8.24]. It is, therefore, convenient to select the state variables x as the angular displacements and velocities of the motors (θ_m), the displacements and velocities of the actuator rod (x_r) and the displacement and velocity of the task space coordinates (Θ). The inputs u of this model, on the other hand, are the actuator currents and the effective frictional forces along the actuator rod. Lastly, the outputs y of the system are chosen as the force output of the force sensors. This means that the state space model can be represented in the form shown in equation [8.29], where the state, input and output vectors are, respectively, defined as $x = [\Theta^T x_r^T \theta_m^T \dot{\Theta}^T \dot{x}_r^T \dot{\theta}_m^T]^T$, $u = [i^T F_{\text{fric}}^T]^T$ and $y = F_{\text{meas}}$.

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \quad [8.29]$$

Although the above state space model represents the system exactly as described by the linearized dynamic equations, it is possible that the real system would have additional delays. For instance, the actuator current is assumed to be an ideal input variable in the above model. However, in reality, the current is in fact controlled using a pulse-width modulation-based motor driver. Consequently, discrepancies will exist between the commanded and actual actuator current. This discrepancy can be modeled as a first-order

low-pass filter with unity gain and a small time constant to reflect the fact that such delay is expected to be relatively low. The introduction of this low-pass filter into the above state space model will require the augmentation of the state vector to include additional states which describes the filter dynamics given in equation [8.30], with i_c being the commanded current issued by the actuator force controller.

$$\tau_{LP} \frac{di}{dt} + i = i_c \quad [8.30]$$

The incorporation of the current dynamics into the above state space model will then result in another state space system with the augmented state vector $x' = [x^T i^T]^T$ and a modified input vector $u' = [i_c^T F_{fric}^T]^T$. This system is represented as equation [8.31], with B_i and B_{fric} being submatrices of the original B matrix which correspond to the current and actuator friction inputs.

$$\dot{x}' = \underbrace{\begin{bmatrix} A & B_i \\ 0_{4 \times 22} & -\frac{1}{\tau} I_4 \end{bmatrix}}_{A'} x' + \underbrace{\begin{bmatrix} 0_{22 \times 4} & B_{fric} \\ \frac{1}{\tau} I_4 & 0_{4 \times 4} \end{bmatrix}}_{B'} u' \quad y = \underbrace{[C \quad 0_{4 \times 4}]}_{C'} x' \quad [8.31]$$

8.3.3.1. Stability analysis of the coupled actuator–sensor–environment system

The stability of the coupled actuator–sensor–environment system under closed-loop force feedback can be evaluated by examining the higher order dynamic models. Even though these models were obtained from a linearized system, they can be utilized to give a good indication on the system behavior around its linearization point. One additional point to keep in mind is that while the model consisting of the decoupled transfer functions assumes that environmental stiffness and damping is proportional to the environmental inertia matrix, no such restriction is placed on the state space model. The state space model can, therefore, be used to give a more general analysis of the coupled system. Having stated this, however, the decoupled transfer functions do have an advantage as they can be easily analyzed in the frequency domain to yield gain margins that is expected to be indicative of the actual gain margins when the proportionality assumption is not severely violated. Therefore, this section considers the stability of the closed-loop system using both models.

8.3.3.2. Stability and robustness analysis in state space

The decoupled transfer functions can only be used if the damping and stiffness matrices are scaled versions of the inertia matrix or if all these matrices share the same decoupled directions. To study more general environments, a state space model can be used as it does not require the above simplification. The basic state space model considered here is that of the actuator–sensor–environment system with low-pass current filtering as presented in section 8.3.3. Due to the use of a digital controller, the stability of the system is analyzed in discrete time. Again, the `c2d` Matlab® command was used to convert the continuous state space model into its discrete counterpart given in equation [8.32].

$$\begin{aligned} x'_{k+1} &= A'_z x'_k + B'_z u'_k \\ y_k &= C'_z x'_k \end{aligned} \quad [8.32]$$

To incorporate the effect of the proportional-derivative (PD) control action into the state space system, the dynamics of discrete time PD control must be considered. This dynamic relationship can be expressed using the difference equation [8.33], where backward differentiation is done with k indicating the current sample number. Note that the variable $F = y$ is used to represent the force sensor measurement as outputted from the discrete state space model in equation [8.32] while F^* is the resulting force after application of the unity gain PD filter. To allow its integration into the state space model, an alternative but equivalent difference equation as shown in equation [8.34] can be considered.

$$F_k^* = F_k + \frac{Kd}{T}(F_k - F_{k-1}) \quad [8.33]$$

$$\begin{aligned} F_{k+1}^* &= \left(1 + \frac{Kd}{T}\right) F_{k+1} - \frac{K_d}{T} F_k \\ &= \left(1 + \frac{Kd}{T}\right) y_{k+1} - \frac{K_d}{T} y_k \\ &= \left[\left(1 + \frac{Kd}{T}\right) C'_z A'_z - \frac{K_d}{T} C'_z \right] x'_k + \left(1 + \frac{Kd}{T}\right) C'_z B'_z u'_k \end{aligned} \quad [8.34]$$

Equation [8.34] shows that additional states of F_k^* can be added to the discrete state space model and that the state transition of this variable is

purely dependent on other states x'_k and inputs u'_k of the current sample. The augmented state space model is represented in equation [8.35].

$$\begin{bmatrix} x'_{k+1} \\ F^*_{k+1} \end{bmatrix} = \begin{bmatrix} \overbrace{A'_z}^{A'_z} & 0_{26 \times 4} \\ \left(1 + \frac{K_d}{T}\right) C'_z A'_z - \frac{K_d}{T} C'_z & 0_{4 \times 4} \end{bmatrix} \begin{bmatrix} x'_k \\ F^*_{k+1} \end{bmatrix} + \begin{bmatrix} \overbrace{B'_z}^{B'_z} \\ \left(1 + \frac{K_d}{T}\right) C'_z B'_z \end{bmatrix} u'_k \quad [8.35]$$

$$F^*_k = \underbrace{[0_{4 \times 26} \quad I_4]}_{C''_z} x''_k$$

When a feedback law with a constant gain matrix K as shown in equation [8.36] is used on the system, the closed-loop difference equation for the system states becomes equation [8.37]. Computing the eigenvalues of the $A''_{z,cl}$ matrix can then reveal the pole locations of the system and stability can be verified if all poles are located within the unit circle on the z -plane (i.e. the magnitudes of the poles are all less than one).

$$u'_k = -Ky'_k \quad [8.36]$$

$$x''_{k+1} = (A''_z - B''_z K C''_z) x''_k = A''_{z,cl} x''_k \quad [8.37]$$

The above state space model can also be used to test the robust stability of the system when uncertainties are incorporated into the original system model. For the purpose of this analysis, all the uncertainties are assumed to be located in the environment and the actual environmental parameters are taken to be the summation of the nominal parameters with an additional uncertain term. Using the above approach and considering equation [8.26], equation [8.24] can be modified to give equation [8.38], where quantities accented with $\sim$ are used to denote the uncertain terms. The introduction of uncertainties to the task space dynamics can also be graphically represented using the block diagram shown in Figure 8.10.

$$\begin{aligned} (M_e \tilde{M}_e) \ddot{\Theta} + (B_e + \tilde{B}_e) \dot{\Theta} + (K_e + \tilde{K}_e) \Theta &= J^T F_{\text{int}} \\ \Rightarrow (M_e + m_f J^T J + \tilde{M}_e) \ddot{\Theta} + (B_e + b_f J^T J + \tilde{B}_e) \dot{\Theta} \\ &+ (K_e + k_f J^T J + \tilde{K}_e) \Theta = b_f J^T \dot{x}_r + k_f J^T x_r \end{aligned} \quad [8.38]$$

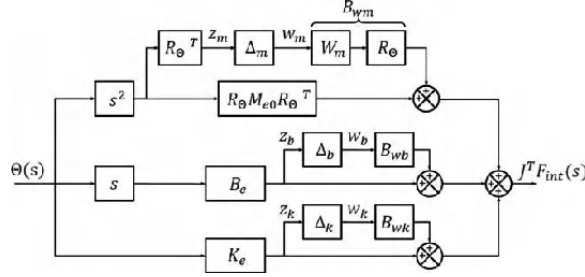


Figure 8.10. Uncertainty formulation of the task space environment

It can be seen that the uncertainty terms $\tilde{M}_e$, $\tilde{B}_e$ and $\tilde{K}_e$ in equation [8.38] were, in turn, divided into several blocks in the uncertainty model structure shown in Figure 8.10. The main reason for this was to make the problem more amenable to application of existing robust stability analysis methods. The relationships between the uncertainty matrices in equation [8.38] and the uncertain terms shown in Figure 8.10 are shown in equations [8.39]–[8.41]. It should be noted that in the remaining equations in this section, the terms Δ_m , Δ_b and Δ_k will be referred to as uncertainty matrices while the terms B_{wm} , B_{wb} and B_{wk} will be referred to as uncertainty weighting matrices.

$$\tilde{M}_e = R_\Theta W_m \Delta_m R_\Theta^T = B_{wm} \begin{bmatrix} \delta_{m1} & 0 & 0 \\ 0 & \delta_{m2} & 0 \\ 0 & 0 & \delta_{m3} \end{bmatrix} R_\Theta^T \quad [8.39]$$

$$\tilde{B}_e = B_{wm} \Delta_b B_e \quad [8.40]$$

$$\tilde{K}_e = B_{wk} \Delta_k K_e \quad [8.41]$$

It can be seen from equations [8.39]–[8.41] that the method used to model uncertainties in the inertia tensor was slightly different compared to that for the environmental stiffness and damping. By assuming that uncertainties in the environmental inertia tensor only affect values of its principal components and that the principal component axes are aligned with the global reference frame at zero task space coordinates, the inertia uncertainty can be written in the form shown in equation [8.39], with Δ_m being a real diagonal matrix and W_m also a diagonal and real weighting matrix. As less information can be assumed about the environmental stiffness and damping, the uncertainties in these parameters were left in a more general form as shown in equations [8.40]–[8.41], where both the uncertainty matrix and the

uncertainty weighting matrix were general 3×3 complex matrices. Note also that the uncertainties for the stiffness and damping were given in multiplicative terms, as opposed to the additive approach used for the inertia parameter. This means that quantities in the W_m matrix must be selected so that they share the same units as that of the inertia tensor, while B_{wb} and B_{wk} are simply dimensionless weightings.

Robust stability is generally analyzed by first grouping all uncertainty matrices into one block and the remaining closed-loop systems into another block. Using the uncertainty assignment presented in Figure 8.10, the overall closed-loop force control system can be put into such a structure and this is shown in Figure 8.11, where the matrix M_{sys} is used to represent the dynamics of the nominal system under closed-loop force control. This particular form will then allow straight forward application of methods such as small gain theorem and structured singular value analysis for the determination of robust stability.

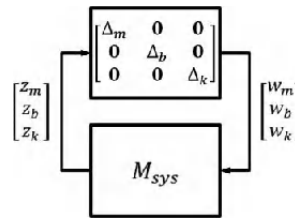


Figure 8.11. Typical representation of systems with uncertainties where the uncertainty block is separated from the overall system

The M_{sys} block shown in Figure 8.11 can be obtained by discretizing an extended version of the state space model given in equation [8.31] and closing the loop with the discrete PD force control law. The extension is required to incorporate additional inputs and outputs to M_{sys} . The inputs of M_{sys} are the outputs of the uncertainty matrices $\varepsilon = [w_m^T w_b^T w_k^T]^T$, while the outputs of M_{sys} are the inputs of the uncertainty matrices $\zeta = [z_m^T z_b^T z_k^T]^T$. It should be noted that the reason behind the disappearance of the commanded current and PD filtered force measurements (the original inputs and outputs of the state space system) from the input and output ports of M_{sys} was the completion of the force feedback loop within M_{sys} . Extension of the state space model, as described above, has led to the representation of M_{sys} as given in equation [8.42]. Construction of the output and direct feed through matrices ($C_{z\zeta}$ and $D_{z\zeta}$) can be done by considering relationships

$z_m = R_\Theta^T \ddot{\Theta}$, $z_b = B_e \dot{\Theta}$ and $z_k = K_e \Theta$. Additionally, since equation [8.43] is true by definition and equation [8.44] can be obtained from equations [8.38] and [8.43], the continuous input matrix for ε (B_e) will only have non-zero entries at rows corresponding to $\ddot{\Theta}$.

$$x''_{k+1} = A''_{z,cl} x''_k + \left[\frac{B_{z\varepsilon}}{(1 + \frac{K_d}{T})C'_z B_{z\varepsilon}} \right] \varepsilon k \quad [8.42]$$

$$\zeta k = C_{z\zeta} x''_k + D_{z\zeta} \varepsilon k$$

$$\tilde{M}_e \ddot{\Theta} + \tilde{B}_e \dot{\Theta} + \tilde{K}_e \Theta = B_{e,\Theta} \varepsilon = [B_{wm} \ B_{wb} \ B_{wk}] \begin{bmatrix} w_m \\ w_b \\ w_k \end{bmatrix} \quad [8.43]$$

$$(M_e + m_f J^T J) \ddot{\Theta} = b_f J^T \dot{x}_r + k_f J^T x_r - (B_e + b_f J^T J) \dot{\Theta} - (K_e - k_f J^T J) \Theta - B_{e,\Theta} \varepsilon \quad [8.44]$$

The small gain theorem is a relatively simple method of analyzing the robust stability of systems which are expressed in the form shown in Figure 8.11. It states that the closed-loop system shown in Figure 8.11 will be stable as long as the product of the maximum singular value for the uncertainty block and that for M_{sys} is less than one over all frequencies. Typically, the uncertainty weighting is “factorized” out of the uncertainty matrix and included in M_{sys} so that the norm of the uncertainty block can be set to unity. With this condition, the stability of the closed-loop system will be fully reliant on the maximum singular value of M_{sys} . As long as M_{sys} is stable, stability of the closed-loop system will be guaranteed if this maximum singular value is less than one. Although relatively simple to compute, the small gain theorem has the disadvantage of being overly conservative [BAT 02, BUE 05] as it allows all elements in the uncertainty block to take on arbitrary values so long as it satisfies the unity norm condition.

An improved analysis method is the structured singular value, which considers the structure within the uncertainty block. Formally, the structure singular value is defined as equation [8.45] [BAT 02], where μ is the structured singular value, $\tilde{\Delta}$ is the set of uncertainty matrix that satisfies the desired structure and $\sigma_{\max}(\Delta)$ denotes the maximum singular value of Δ . In other words, the structured singular value is inversely related to the size/norm

of the smallest uncertainty matrix Δ (which shares the same structure as $\tilde{\Delta}$) that can cause the system M_{sys} to become unstable by making $\det(I - M_{\text{sys}}\Delta) = 0$. On the basis of this definition, a smaller structured singular value indicates higher robust stability. When the uncertainty weightings have been selected to allow a maximum norm of one for Δ , it can be seen that as long as μ is smaller than one, the system will be robustly stable for all possible systems, which falls within the chosen uncertainty bounds. It should be noted, however, that analytical solutions to μ can only be obtained for certain special cases and it is, in general, estimated numerically. The μ function in Matlab[®] is one such function and it was used in this work.

$$\mu(M_{\text{sys}}) = \frac{1}{\inf_{\Delta \in \tilde{\Delta}} \{\sigma_{\max}(\Delta) \mid \det(I - M_{\text{sys}}\Delta) = 0\}} \quad [8.45]$$

As M_{sys} is frequency dependent, computation of the structured singular value will need to be done at all frequencies and the largest result should be returned. To do so, the frequency response of M_{sys} must be obtained. By considering the discrete state space model in equation [8.42], the z -transform of ζ can be related to the z -transform of ε through equation [8.46]. Since $z = e^{sT}$ and the substitution of $s = j\omega$ is used to obtain the frequency response of transfer functions, the frequency response $M_{\text{sys}}(j\omega)$ is given by equation [8.47].

$$M_{\text{sys}}(z) = \frac{\zeta(z)}{\varepsilon(z)} = \left\{ C_{z\zeta} (zI - A_{z,cl}^*)^{-1} \left[\frac{B_{ze}}{(1 + \frac{k_d}{T})C_z' B_{ze}} \right] + D_{z\zeta} \right\} \quad [8.46]$$

$$M_{\text{sys}}(j\omega) = \frac{\zeta(j\omega)}{\varepsilon(j\omega)} = \left\{ C_{z\zeta} (e^{j\omega T} I - A_{z,cl}^*)^{-1} \left[\frac{B_{ze}}{(1 + \frac{k_d}{T})C_z' B_{ze}} \right] + D_{z\zeta} \right\} \quad [8.47]$$

The result in equation [8.47] can then be computed on a frequency grid to produce $M_{\text{sys}}(j\omega)$ at discrete intervals. This information can then be passed into the μ function to obtain an upper and lower bound on $|\mu(j\omega)|$ over the considered frequencies. The maximum of the upper bound of $|\mu(j\omega)|$ can then be used as an estimate for the structured singular value of M_{sys} . The result of

the structured singular value analysis of the system linearized about the zero task space coordinates is plotted in Figure 8.12 to give an indicative example of $|\mu(j\omega)|$. The highest peak, which is considered to be the structured singular value for M_{sys} , is also marked on the plot. In this chapter, the frequency grid used to compute $M_{sys}(j\omega)$ spans from 10^{-3} to π/T rad/s (the frequency beyond which aliasing will occur in the discrete control system). The final frequency grid used is uniformly spaced on the logarithmic frequency scale with 200 intervals. The choice of the number of intervals was selected by trial and error and is done by gradually decreasing the number of frequency intervals while observing the peak values of $|\mu(j\omega)|$ obtained from the μ function. The smallest number of intervals, which preserves the value of the peaks, was chosen for use in the final analysis. For simplicity, the uncertainty weighting matrices were not chosen to be frequency dependent and were simply selected as scaled identity matrices.

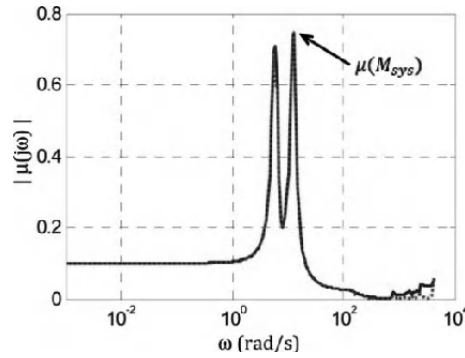


Figure 8.12. The structured singular values of the system at different frequencies. Solid line indicates the upper bound and dotted line indicates the lower bound

8.3.4. Proposed actuator force controller

On the basis of the above analyses, a gain-scheduled actuator joint force controller was proposed and implemented on the ankle rehabilitation robot as this will give a more optimal performance. The structure of this controller is shown in Figure 8.13, where F_c is a vector of the commanded forces and F_{meas} is a vector of forces as provided by the force sensors. K is the gain matrix, which is given by equation [8.48], where U'_m and v_0 are the output basis

vectors of M_e that can be obtained from the singular value decomposition of M_e . Furthermore, $\text{diag}(\cdot)$ is a function that forms a diagonal matrix using its argument and k_i are the controller gains applied to the i th basis vector, with $i=0$ referring to the gain along the null space direction and $i=1,2,3$ referring to gains along decoupled directions with the first, second and third largest singular values, respectively.

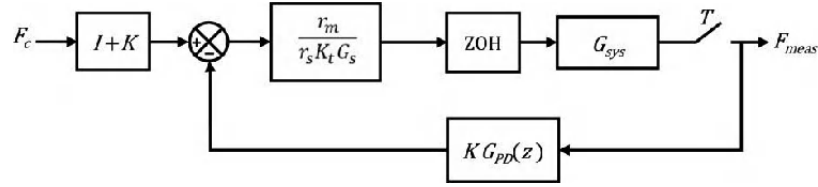


Figure 8.13. Structure of the final actuator force control law

$$K = [U'_m \ v_0] \text{diag}([k_1 \ k_2 \ k_3 \ k_0]) [U'_m \ v_0]^T \quad [8.48]$$

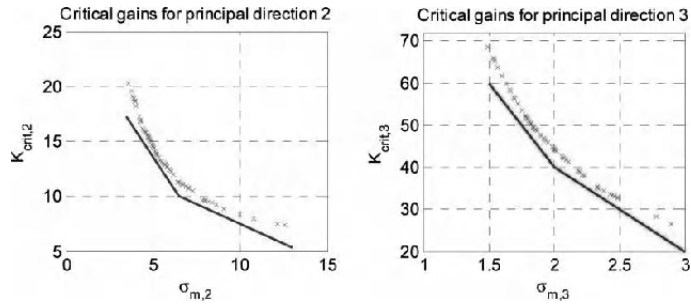


Figure 8.14. Linear piecewise functions used for gain scheduling in different decoupled directions of M_e'

Gain scheduling for the controller gains in two of the three non-null decoupled directions was done using piecewise linear functions fitted below the critical gain values. These relationships are shown as solid lines in Figure 8.14, together with the critical gain values for comparison. The remaining non-null direction, on the other hand, was assigned a constant gain since the critical gain variation over the range of singular values considered is relatively small. In addition, as the null direction is not influenced by the configuration of the manipulator, it is assigned a constant gain that is below its gain margin.

A computational ankle model developed in [TSO 10] was used to obtain a more realistic operating environment for the ankle rehabilitation robot. The environmental stiffness experienced during passive ankle motion was estimated from the static torque observed at different ankle and subtalar joint displacements. These torque profiles were then numerically differentiated with respect to the ankle and subtalar displacements to produce the stiffness matrix K_{as} , which is a 2×2 matrix. This matrix was subsequently transformed into the manipulator task space to give a 3×3 stiffness matrix k_e . On the other hand, the environmental damping matrix B_e was assumed to have the form shown in equation [8.49], with both a constant and a variable component. In equation [8.49], b_e is a scalar constant that gives a base level of damping while $\gamma \ll 1$ is a proportionality constant between the damping and stiffness parameters. The variable component had been included to introduce additional damping in a proportional manner to the ankle stiffness and is mainly used to reduce the damping factor experienced at foot orientations with very high stiffness.

$$B_e = b_e I + \gamma K_e \quad [8.49]$$

A structured singular value analysis of the proposed actuator force controller was carried out to determine its stability and robustness properties. This analysis was carried out at discrete points in the task space that corresponds to different combinations of ankle and subtalar joint displacements within a certain range ($\pm 40^\circ$ at 4° interval for the ankle displacement and $\pm 30^\circ$ at 3° interval for the subtalar displacement). At each of these task space coordinates, the environmental stiffness and damping matrices were obtained and used to form the nominal external environment. The uncertainty weighting matrices used in this analysis are given in Table 8.4. The uncertainties considered represent about 5% of the total nominal mass parameter and 10% variations in the stiffness and damping parameters. Note that the environmental mass includes that of the manipulator and as a result of that, 5% uncertainty in the total nominal mass parameters actually relates to a higher proportion (about 10%) of the foot rotational inertia according to estimates derived from [NAS 08].

Uncertainty weighting	B_{wm}	B_{wb}	B_{wk}
Value/expression	$0.002R_\theta \text{ kgm}^2$	$0.1I_3$	$0.1I_3$

Table 8.4. *Uncertainty weighting matrices used in the robust stability analysis*

Stability analysis of the $A''_{z,cl}$ matrix in M_{sys} had shown that all system poles are located within the unit circle on the z -plane for the foot configurations tested, thus proving system stability and ensuring that structure singular values can be used to evaluate the robust stability of the system. The results of the structured singular value analyses are summarized in Figure 8.15, where the computed values of $\mu(M_{sys})$ are plotted over their corresponding ankle and subtalar joint displacements. It can be seen in this figure that all the structured singular values are less than one, thus indicating that the system will remain stable as long as the environmental uncertainties remain within the prescribed bounds.

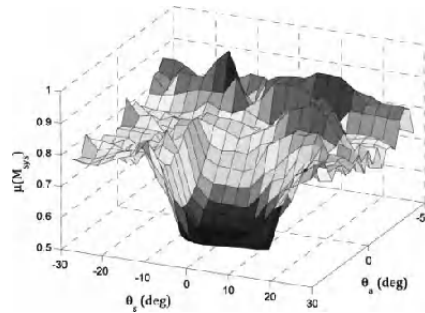


Figure 8.15. Values of $\mu(M_{sys})$ computed over a range of foot orientations defined by the ankle (θ_a) and subtalar (θ_s) joint displacements

8.4. Experimental results

Several experimental trials were carried out on the ankle rehabilitation robot developed in this research to evaluate the effectiveness of the proposed actuator force controller. These experiments can be classified into two groups: one to highlight the effects of different force controllers on the system stability and another to evaluate and compare the force control performance of different controllers in executing tasks required by the rehabilitation exercises.

8.4.1. Stability experiment

As one of the main motivations behind the development of the proposed force controller is to improve the system stability, one of the experimental trials carried out in this work was centered on demonstrating the stability

improvement brought on through the application of the proposed control scheme. This experimental trial involved the operation of the ankle rehabilitation robot under pure force control, where the force commands were selected by requiring that the vertical component of the null space forces be summed to a desired value to provide support for the user's lower limb, while all remaining non-null space forces are set to be zero. The subject was then prompted to move his or her foot freely in primarily the plantar-/dorsiflexion direction. This experiment was done for four different actuator force controllers, which are listed and described in Table 8.5. The results of this experiment are presented in Figure 8.16. Note that only the forces measured along one of the actuators are shown for brevity, and forces along other directions behave in a similar manner.

Controller	Description
P control, uniform gain of 10	Measured force is fed directly into a disturbance observer, the gain matrix K in [8.48] is obtained by selecting $k_1 = k_2 = k_3 = k_4 = 10$
PD control, uniform gain of 16	Measured force is fed into a disturbance observer through a proportional-derivative filter $G_{PD}(z)$, the gain matrix K in [8.48] is obtained by selecting $k_1 = k_2 = k_3 = k_4 = 16$
Proposed controller	Measured force is fed into a disturbance observer through a proportional-derivative filter $G_{PD}(z)$, the gain matrix K in [8.48] is obtained by observing the gain margins given in Figure 8.14
Proposed controller + minimum gain of 14	Measured force is fed into a disturbance observer through a proportional-derivative filter $G_{PD}(z)$, the gain matrix K in [8.48] is obtained by setting $k_1 = k_0 = 14$, while k_2 and k_3 are obtained by observing the gain margins given in Figure 8.14

Table 8.5. *Actuator force controllers considered in the stability experiments*

From the experiments, motions obtained using the first two controllers showed clear signs of instability with significant levels of oscillation. This is backed up by the actuator force measurements given in Figure 8.16. Since the gains of the uniform gain controllers were selected to be the lowest gains that will result in perceptible oscillations in the robot, it can be seen that the addition of the proportional derivative term in the feedback loop does indeed improve the stability of the system and allow better performance through

application of larger controller gains. Furthermore, Figure 8.16 also shows that although some of the gains applied along certain decoupled directions of the third and fourth controllers considered were larger in value than that of the second uniform gain controller, these controllers remained stable during the operation. This supports the idea that there are directions that are less stable and these directions ultimately limit the maximum gain that can be applied in a controller with uniform gains. Consequently, the important conclusion that can be drawn from this experiment is that when actuators are independently controlled with its own disturbance observer (the uniform gain control approach), the maximum performance achievable would be limited due to the upper gain limit imposed by the system stability. However, when the coupling introduced by the manipulator kinematics and inertia is taken into account, it is possible to manage stability through application of different gains along different decoupled directions of M'_e . This allows higher gains in more stable directions, and thus results in an improved overall performance.

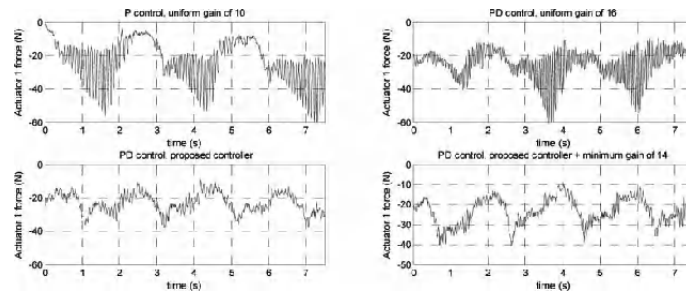


Figure 8.16. Forces measured along actuator 1 using different actuator force controllers

8.4.2. Experiments for performance evaluation

In addition to the stability experiment, further trials were also carried out using both a uniform gain controller and the proposed controller to illustrate the performance improvements afforded through incorporation of the coupling information into the controller. These experiments were again done on the developed ankle rehabilitation robot and involve two main tasks that are considered important for implementation of ankle rehabilitation exercises and will test the force controllers. The first task is the ability to maximize the backdrivability of the robot by commanding zero task space moments (such as that done in the stability experiment) and the second task is the explicit

control of the robot–user interaction moment. Both tasks were carried out using a uniform gain actuator force controller with a gain of five, and the proposed controller (note that the gain of five is also used along the output basis vector of M'_e which has the largest singular value). The results for the first task described above are summarized in Figure 8.17 and the results obtained from the second task are presented in Figures 8.18 and 8.19.

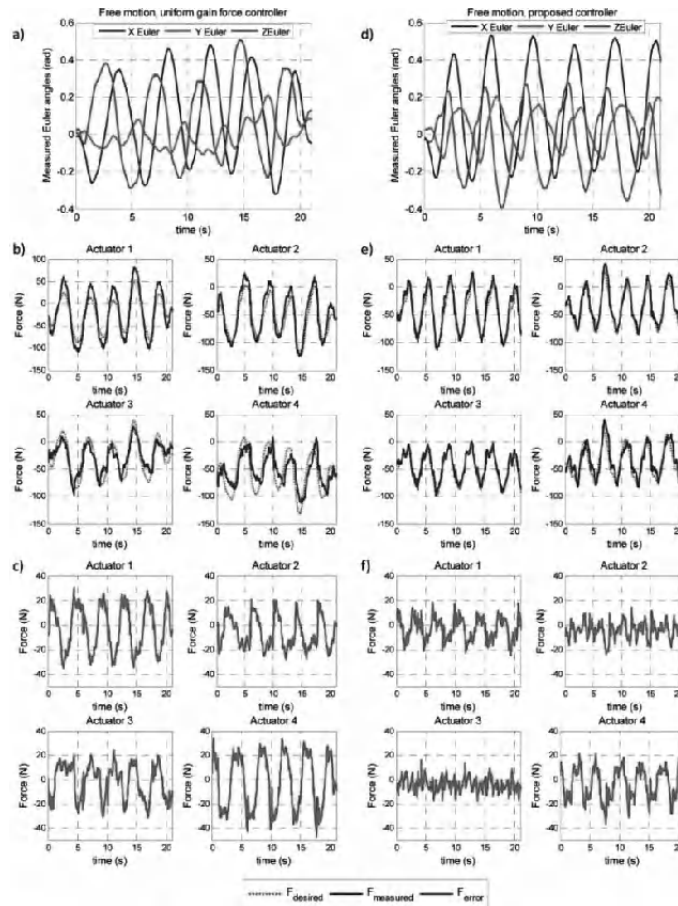


Figure 8.17. Experimental results obtained during free motion of the user's foot on the ankle rehabilitation robot. The Euler angle trajectories for the uniform gain controller are shown in a) while the associated actuator forces and force errors are given in b) and c). Similarly, the motion trajectories obtained using the proposed controller are given in d) and its associated forces and force errors are shown in e) and f)

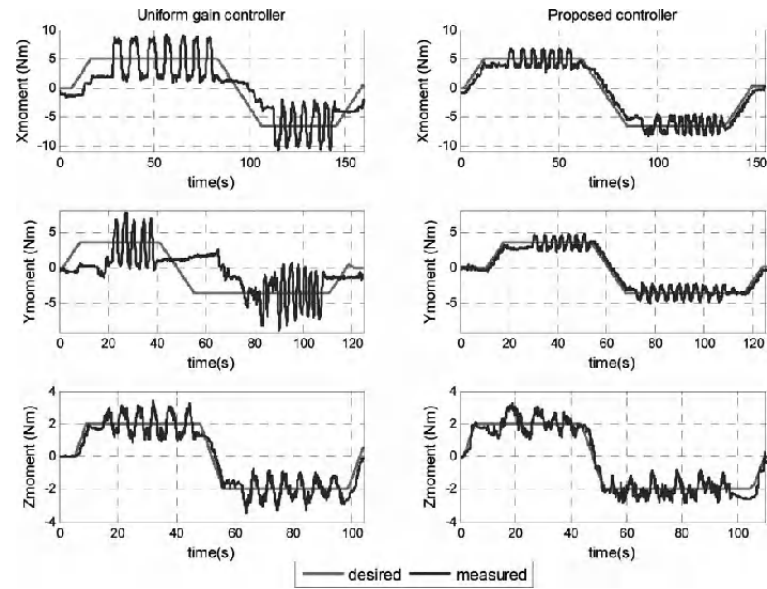


Figure 8.18. *Desired and measured ankle moments about the ankle as obtained from the torque control experiment*

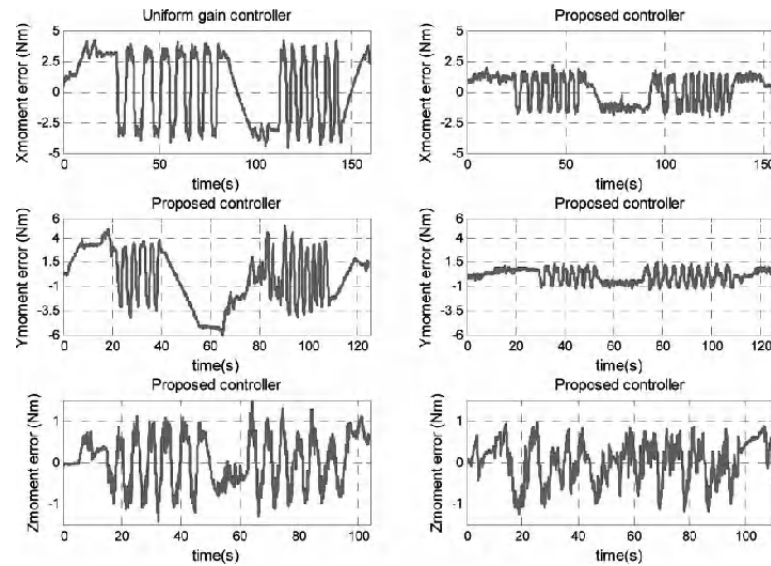


Figure 8.19. *Moment errors as obtained from the torque control experiment*

It can be seen from the results obtained in the free motion trials that utilization of the proposed control scheme over the uniform gain controller can significantly reduce the actuator force errors, with some of the actuators experiencing a force error reduction of about 50%. Comparison of Figures 8.17(a) and (d) also suggests that the force-proposed controller is more capable in terms of maximizing the compliance of the ankle rehabilitation robot since it appears that motion recorded using the proposed controller is of larger amplitudes and velocities relative to that of the uniform gain controller.

Inspection of results from the torque control trials also showed similar trends, where moment tracking capability of the robot is significantly improved in the X and Y directions when the proposed controller is used over the uniform gain controller. The moment performance along the Z-direction, however, is similar between the two controllers. This can be explained by how the task space moments are related to the actuator forces through the manipulator Jacobian transpose J^T . Singular decomposition of J^T can be used to show that the output basis vector of J^T with the most influence on the Z-direction is linked to an input basis vector that is closely aligned with the basis vector of K with the smallest gain. Because this smallest gain is also of the same magnitude as the gain used in the uniform gain controller, it is not surprising that similar moment errors were observed along the z -direction. Similarly, the X and Y task space moments can also be found to be more closely linked to the directions where gains k_2 and k_3 are applied in the proposed controller. This further agrees with the observation that smaller moment errors were recorded in the Y-direction.

The results of the torque control trials have suggested that the moment errors are large, even with the use of the proposed controller. It should be noted, however, that this error is mainly caused by frictional effects and does not vary significantly with the amplitude of the commanded torque levels. This is shown in Figure 8.20, where the moment regulation performance of the robot in the X-direction was tested at three different levels. It is clear from the moment error plot that the magnitude of the errors remained relatively constant regardless of the value of the reference moment. This implies that the robot is not capable of realizing the desired moment in a very precise manner. Given that low moment commands are used mainly to improve the robot backdrivability, the above results mean that an effective frictional moment of approximately 1.5 Nm is to be expected on the robot. Although not ideal, this is considered to be acceptable for this application, as the user should be able to easily overcome such resistance. For tasks

involving larger moment commands such as strengthening exercises, the moment error will become less significant and will not severely degrade the performance of the robot.

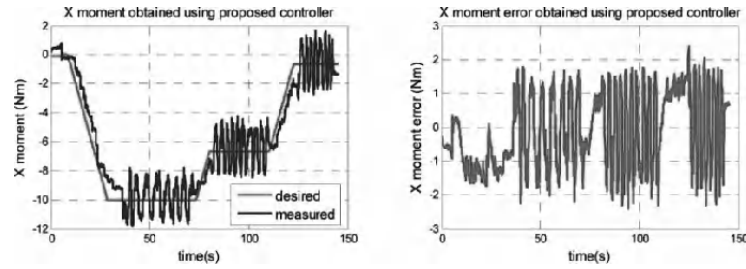


Figure 8.20. Experimental results obtained by using the proposed controller in regulation of the X interaction moment at three different levels

8.5. Concluding remarks

This chapter detailed the development of a MIMO actuator force controller for the ankle rehabilitation robot that incorporates the user into the system. The coupling between actuator forces was first identified and an initial design of a disturbance observer-based decoupling force control scheme that considers the kinematic and inertial characteristics of the robot was presented. Although this decoupling control scheme cannot be implemented directly on the robot due to stability issues originating from the presence of unmodeled dynamics, it has provided valuable insights into the coupled force dynamics by highlighting the possibility of transforming the force control problem into a separate force coordinate frame whose bases are spanned by the output basis vectors or decoupled directions of the coupling matrix D .

Using this transformation, further analyses were carried out on a higher order model of the actuator, which includes the actuator and force sensor compliance. By applying certain simplifying assumptions, the force control problem was completely decoupled along these decoupled directions. On the basis of these analyses, an additional proportional derivative filter was included into the disturbance observer and gain margins along these decoupled directions were established for different orientations of the robot end-effector. It was found that the gain margins vary rather significantly among different decoupled directions as well as across different end-effector orientations of the ankle rehabilitation robot. This had led to the proposal of a gain-scheduled MIMO actuator force controller in this research.

Experiments have shown the proposed method to be effective in improving the performance of the force controller over the conventional uniform gain approach (which is also the approach where each actuator is controlled independently from one another). Additionally, as the computed gain margins allow partial decoupling of the actuator–robot system, the condition number of the effective end-effector inertia is also significantly reduced in the proposed control scheme, thus making the backdrivability of the robot more uniform in different directions.

8.6. Bibliography

- [ACC 07] ACC, Accident Compensation Corporation, New Zealand, retrieved 24 October 2007. Available at <http://www.acc.co.nz/for-providers/resources/WCMZ002647?ssSourceNodeId=4249&ssSourceSiteId=149>.
- [AN 87] AN C.H., HOLLERBACH J.M., “Dynamic stability issues in force control of manipulators”, *IEEE International Conference on Robotics and Automation*, 1987.
- [BAT 02] BATES D.G., POSTLETHWAITE I., “The structured singular value and mu-analysis”, in FIELDING C., VARGA A., BENNANI S., SELIER M. (eds), *Advanced Techniques for Clearance of Flight Control Laws*, Springer, Berlin, pp. 37–55, 2002.
- [BUE 05] BUERGER S.P., Stable, high-force, low impedance robotic actuators for human-interactive machines, PhD Dissertation, MIT, 2005.
- [BUE 06] BUERGER S.P., HOGAN N., “Relaxing passivity for human-robot interaction”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006.
- [BUE 07] BUERGER S.P., HOGAN N., “Complementary stability and loop shaping for improved human–robot interaction”, *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 232–244, 2007.
- [DUL 85] DUL J., JOHNSON G.E., “A kinematic model of the human ankle”, *Journal of Biomedical Engineering*, vol. 7, no., pp. 137–143, 1985.
- [GIR 01] GIRONE M., BURDEA G., BOUZIT M., *et al.*, “Stewart platform-based system for ankle telerehabilitation”, *Autonomous Robots*, vol. 10, pp. 203–212, 2001.
- [HAB 06] HABIB M.K., “Mechatronics engineering the evolution, the needs and the challenges”, *32nd Annual Conference of IEEE Industrial Electronics Society*, 2006.
- [HAB 08] HABIB M.K., “Interdisciplinary mechatronics: problem solving, creative thinking and concurrent design synergy”, *International Journal of Mechatronics and Manufacturing Systems*, vol. 1, no. 1, pp. 264–269, 2008.

- [HAR 95] HARWIN W.S., RAHMAN T., FOULDS R.A., "A review of design issues in rehabilitation robotics with reference to North American research", *IEEE Transactions on Rehabilitation Engineering*, vol. 3, no. 1, pp. 3–13, 1995.
- [HER 02] HERTEL J., "Functional anatomy, pathomechanics, and pathophysiology of lateral ankle instability", *Journal of Athletic Training*, vol. 37, no. 4, pp. 364–375, 2002.
- [HES 03] HESSE S., SCHMIDT H., WERNER C., *et al.*, "Upper and lower extremity robotic devices for rehabilitation and for studying motor control", *Current Opinion in Neurology*, vol. 16, pp. 705–710, 2003.
- [HOG 87] HOGAN N., "Stable execution of contact tasks using impedance control", *IEEE International Conference on Robotics and Automation*, 1987.
- [HOG 05] HOGAN N., BUERGER S.P., "Impedance and interaction control", in KURFESS T. (ed.), *Robotics and Automation Handbook*, CRC Press, New York, 2005.
- [INM 76] INMAN V.T., *The Joints of the Ankle*, Williams and Wilkins, Baltimore, 1976.
- [ISM 68] ISMAN R.E., INMAN V.T., Anthropometric studies of the human foot and ankle, Technical Report 58, The Laboratory, Biomechanics Laboratory, University of California, San Francisco and Berkeley, p. 33, May 1968.
- [KAT 06] KATSURA S., MATSUMOTO Y., OHNISHI K., "Analysis and experimental validation of force bandwidth for force control", *IEEE Transactions on Industrial Electronics*, vol. 53, no. 3, pp. 922–928, 2006.
- [KAT 08] KATSURA S., IRIE K., OHISHI K., "Wideband force control by position-acceleration integrated disturbance observer", *IEEE Transactions on Industrial Electronics*, vol. 55 no. 4, pp. 1699–1706, 2008.
- [KEA 90] KEARNEY R.E., WEISS P.L., MORIER R., "System identification of human ankle dynamics: intersubject variability and intrasubject reliability", *Clinical Biomechanics*, vol. 5, pp. 205–217, 1990.
- [KEM 99] KEMPF C.J., KOBAYASHI S., "Disturbance observer and feedforward design for a high-speed direct-drive positioning table", *IEEE Transactions on Control Systems Technology*, vol. 7 no. 5, pp. 513–526, 1999.
- [KON 08] KONG K., TOMIZUKA M., MOON H., *et al.*, "Mechanical design and impedance compensation of SUBAR (Sogang University's Biomedical Assist Robot)", *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Xi'an, China, 2008.
- [KRE 00] KREBS H.I., VOLPE B.T., AISEN M.L., *et al.*, "Increasing productivity and quality of care: robot-aided neuro-rehabilitation", *Journal of Rehabilitation Research and Development*, vol. 37, no. 6, pp. 639–652, 2000.

- [LIN 08] LIN C.-C.K., JU M.-S., CHEN S.-M., *et al.*, “A specialized robot for ankle rehabilitation and evaluation”, *Journal of Medical and Biological Engineering*, vol. 28, no. 2, pp. 79–86, 2008.
- [LIU 06] LIU G., GAO J., YUE H., *et al.*, “Design and kinematics simulation of parallel robots for ankle rehabilitation”, *IEEE International Conference on Mechatronics and Automation*, Luoyang, China, 2006.
- [MAT 02] MATTACOLA C.G., DWYER M.K., “Rehabilitation of the ankle after acute sprain or chronic instability”, *Journal of Athletic Training*, vol. 37, no. 4, pp. 413–429, 2002.
- [MER 06] MERLET J.-P., *Parallel Robots*, Springer, The Netherlands, 2006.
- [MIT 03] MITTAL R.K., NAGRATH I.J., *Robotics and Control*, Tata McGraw-Hill Education, New Delhi, 2003.
- [MUR 93] MURAKAMI T., YU F., OHNISHI K., “Torque sensorless control in multidegree-of-freedom manipulator”, *IEEE Transactions on Industrial Electronics*, vol. 40, no. 2, pp. 259–265, 1993.
- [NAS 08] NASA, National Aeronautics and Space Administration, retrieved 14 July 2008. Available at <http://msis.jsc.nasa.gov/sections/section03.htm>.
- [PAR 98] PARENTEAU C.S., VIANO D.C., PETIT P.Y., “Biomechanical properties of human cadaveric ankle-subtalar joints in quasi-static loading”, *Journal of Biomechanical Engineering*, vol. 120, pp. 105–111, 1998.
- [REI 04] REINKENSMAYER D.J., EMKEN J.L., CRAMER S.C., “Robotics, motor learning, and neurologic recovery”, *Annual Review of Biomedical Engineering*, vol. 6, pp. 497–525, 2004.
- [RIE 05] RIENER R., FREY M., BERNHARDT M., *et al.*, “Human-centered rehabilitation robotics”, *IEEE International Conference on Rehabilitation Robotics*, 2005.
- [ROY 07] ROY A., KREBS H.I., PATTERSON S.L., *et al.*, “Measurement of human ankle stiffness using the anklebot”, *International Conference on Rehabilitation Robotics*, 2007.
- [SAF 99] SAFRAN M.R., ZACHAZEWSKI J.E., BENEDETTI R.S., *et al.*, “Lateral ankle sprains: a comprehensive review part 2: treatment and rehabilitation with an emphasis on the athlete”, *Medicine & Science in Sports & Exercise*, vol. 31, no. 7, pp. S438–S447, 1999.
- [SAG 09] SAGLIA J.A., TSAGARAKIS N.G., DAI J.S., *et al.*, “A high-performance redundantly actuated mechanism for ankle rehabilitation”, *The International Journal of Robotics Research*, vol. 28, no. 9, pp. 1216–1227, 2009.
- [SAG 10] SAGLIA J.A., TSAGARAKIS N.G., DAI J.S., *et al.*, “Control strategies for ankle rehabilitation using a high performance ankle exerciser”, *IEEE International Conference on Robotics and Automation*, 2010.

- [SIE 88] SIEGLER S., CHEN J., SCHNECK C.D., “The three-dimensional kinematics and flexibility characteristics of the human ankle and subtalar joints-part I: kinematics”, *Journal of Biomechanical Engineering*, vol. 110, pp. 364–373, 1988.
- [SUN 07] SUN J.G., GAO J.Y., ZHANG J.H., *et al.*, “Teaching and playback control system for parallel robot for ankle joint rehabilitation”, *IEEE International Conference on Industrial Engineering and Engineering Management*, 2007.
- [SYR 08] SYRSELOUDIS C.E., EMIRIS I.Z., “A parallel robot for ankle rehabilitation-evaluation and its design specifications”, *IEEE International Conference on Bioinformatics and Bioengineering*, 2008.
- [TEJ 00] TEJIMA N., “Rehabilitation robotics: a review”, *Advanced Robotics*, vol. 14, no. 7, pp. 551–564, 2000.
- [TRU 10] TRUONG D.Q., AHN K.K., NAM D.N.C., *et al.*, “Estimation of bending behavior of an ionic polymer metal composite actuator using a nonlinear black-box model”, *International Conference on Control, Automation and Systems (ICCAS)*, 2010.
- [TSA 99] TSAI L.-W., *Robot Analysis: The Mechanics of Serial and Parallel Manipulators*, John Wiley & Sons, New York, NY, 1999.
- [TSO 10] TSOI Y.H., Modelling and adaptive interaction control of a parallel robot for ankle rehabilitation, PhD Thesis, University of Auckland, 2010.
- [WAN 04] WANG Y., XIONG Z., DING H., *et al.*, “Nonlinear friction compensation and disturbance observer for a high-speed motion platform”, *International Conference on Robotics & Automation*, New Orleans, LA, 2004.
- [YIN 05] YING N., KIM W., “Determining dual Euler angles of the ankle complex in vivo using ‘flock of birds’ electromagnetic tracking device”, *Journal of Biomechanical Engineering*, vol. 127, pp. 98–107, 2005.
- [YOO 05] YOON J., RYU J., “A novel reconfigurable ankle/foot rehabilitation robot”, *IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005.
- [YOO 06] YOON J., RYU J., LIM K.-B., “Reconfigurable ankle rehabilitation robot for various exercises”, *Journal of Robotic Systems*, vol. 22, pp. S15–S33, 2006.

Chapter 9

Performance Evaluation of a Probe Climber for Maintaining Wire Rope

In recent years, with barrier-free public facilities, elevators have come to be installed in stations and on footbridges. The shortage of maintenance technicians has become a social problem with the increase in elevator. In addition, an inspection of the elevator and the amusement machine that have been suspended by wire rope is often overlooked since it depends on visual inspection. For such a problem, by making a self-driving probe climber work efficiently, the safety of workers and an improvement in the quality of maintenance can be expected. In this chapter, we developed a self-driving probe climber to detect the wear and corrosion damage of the wire rope at an early stage. It also provides the results of their verification for damage detection performance of wire rope and driving performance.

9.1. Introduction

As public facilities have become barrier-free in recent years, elevators have been introduced in low-rise buildings, in station buildings and on pedestrian bridges, and the amount of elevator maintenance is approaching one million units. However, a shortage of engineers who possess proper maintenance techniques has become an issue. In addition, visual inspections are mostly conducted for double-track normal ropeway stays as well as

Chapter written by Akihisa TABATA, Emiko HARA and Yoshio AOKI.

amusement facilities for which fixed wire ropes are used for load bearing. Many of those maintenance inspections involve risks, such as high-place work. A countermeasure against these is to develop an automatic running inspection device having a function of early detection of the wire breakages, environmental degradation and abnormal abrasion of wire rope, by which secured safety of workers, streamlined inspection work and improved quality in maintenance inspections can be expected.

In this study, in order to develop a remote maintenance inspection system of wire rope for structures, an inspection device was first mounted for evaluating structural integrity and an elevation mechanism (probe climber) was fabricated that was capable of even vertical automatic movements. Then, through wire rope elevation experiments, appropriate conditions for friction drive to enable stable elevation with less energy consumption were examined. Magnetic flux leakage (MFL) flaw detect method is currently common for evaluating wire rope integrity. However, the method has limitations, such as the inspection device's scanning speed and detection accuracy for wire breakages and corrosion in rope trough and bottom areas has become an issue. This study considers the detection accuracy after configuring a sensor system of phased-array type Hall elements, which is more compact and accurate in detection than conventional products. It also describes the evaluation results through experiments for identifying the locations of damage/corrosion in conjunction with a drive control system of the automatic elevation device and makes a case analysis of damage/corrosion patterns by using cognition and judgment techniques.

9.2. Optimize friction drive conditions using a prototype probe climber

Figure 9.1 shows an automatic probe climber made as the first prototype. The designed probe climber has an elevation mechanism where a wire rope is sandwiched between drive wheel and idler wheel, the latter of which is pressed against the wire rope in order to secure sufficient frictional force required to elevate the machine body. As shown in Figure 9.2, push mechanisms with screws and springs were fabricated for adjusting the pushing force of the idler wheel, and the impact of the adjustment made on the elevation performance and power consumption was examined [HAL 12].

The pushing force was adjusted relative to the total weight of the climber, in the range from the lower limit for securing the minimum frictional force required for vertical elevation to the upper limit at which elevation is no longer possible due to the high load of the pushing force. In the case of the

push mechanism with screws, the screws were tightened for each pitch, whose axial force was measured and managed by a load cell. For the push mechanism with springs, frictional force adjustment was made possible by retrofitting springs with different spring constants.

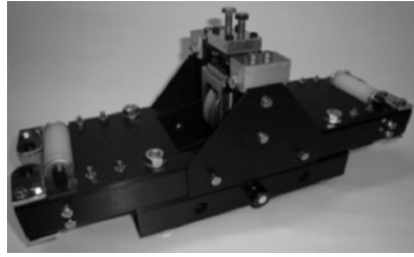


Figure 9.1. *Prototype model of probe climber*

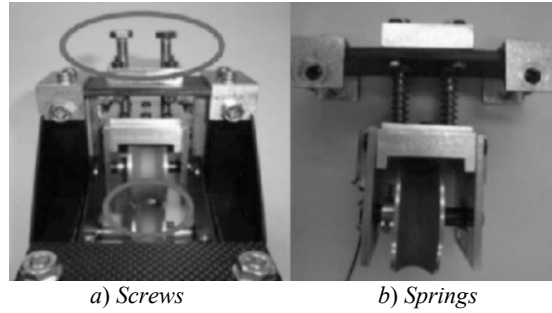


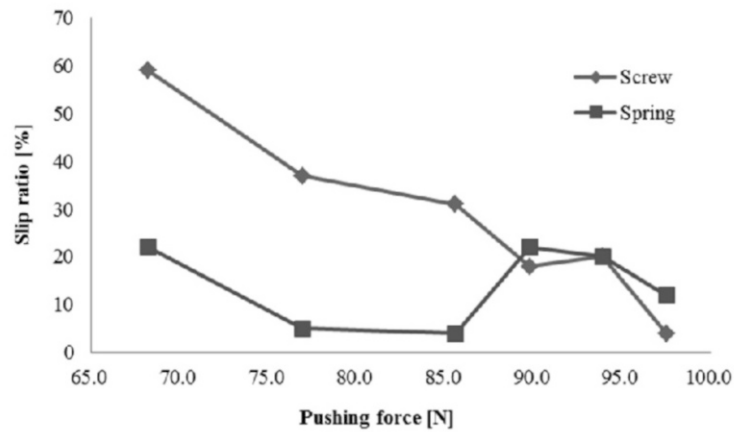
Figure 9.2. *Friction force adjustment mechanism*

During the elevation experiments, the pushing forces were changed: 68.2 N (lower limit), 76.9, 85.6, 89.8, 93.9, 97.5 N (upper limit). Measurements were made for elevation speed, power consumption, three-axis acceleration and the rotational speed of drive shaft and idler shaft. Besides, the slip ratio of the drive wheel was calculated by the following formula:

$$S = \frac{v_w - v_v}{v_w} \quad [9.1]$$

where $S[\%]$ is slip ratio, v_w is drive wheel speed and v_v is idler wheel speed. Table 9.1 and Figure 9.3 show the relationship between the slip ratio calculated by formula [9.1] and the pushing force.

Pushing force (N)	Screw				Spring			
	Slip ratio (%)	Current (A)	Time (s)	Vibration (m/s^2)	Slip ratio (%)	Current (A)	Time (s)	Vibration (m/s^2)
68.2	59	5.28	9.8	2.218	22	5.40	10.7	1.874
76.9	37	5.49	9.8	2.241	5	5.48	9.6	1.795
85.6	31	5.66	10.2	2.315	4	5.50	12.3	2.253
89.8	18	5.71	11.0	2.301	22	5.58	12.5	2.275
93.9	20	5.87	10.9	2.223	20	5.68	12.6	2.270
97.5	4	5.99	11.6	2.520	12	5.68	13.3	2.790

Table 9.1. Climbing performance**Figure 9.3.** Change of the slip rate with the pushing force

It was found that in both mechanisms, the slip ratio would decrease, when the pushing force was increased from 68.2 to 85.6 N. The tendency showed that when the push mechanism with springs was adopted, its slip ratio would become smaller than that of the one with screws. Conceivably, the slip ratio became bigger in the screw pushing mechanism, where the idler wheel range of motion (ROM) is smaller than that in the spring pushing mechanism, thus the former could not follow changes in the wire rope diameter during the elevation. The vibration magnitude and the slip ratio in Table 9.1 indicated that even the spring pushing mechanism would have similar slip ratio to the screw pushing mechanism's, when the springs were compressed in the vicinity of the closed length. Furthermore, in the case of smaller pushing force, the spring pushing mechanism has a smaller vibration than the screw

pushing mechanism, thus vibration suppression effect can be expected when ROM is sufficiently secured in the spring pushing mechanism. Partly because the wire rope diameters in damaged and worn areas have locally changed, the probe climber should be designed so that the spring pushing mechanism may be proper with the spring ROM sufficiently secured.

9.3. Impact of different surface friction materials for friction pulley made on elevation performance

Consideration was made for inserting friction material onto the surface of the drive wheel and the idler wheel in order to cause no damage to the rope surface during wire rope maintenance inspection and ensure improvement in elevation performance. In this regard, a simple friction measuring device was made so as to measure coefficients of kinetic friction (μ) of various types of friction materials shown in Table 9.2, when they were pressed against a synthetic belt. Figure 9.4 shows the relationship between the pushing force and coefficients of kinetic friction. It was found that polychloroprene rubber among the friction material types is a suitable friction material with a relatively large kinetic friction coefficient, even though the hardness is 70. Also, the kinetic friction coefficient (μ) in any of the material types would be smaller, as the pushing force increased [SHA 12]. It has been verified that increased pushing force will add running load to the climber and increase energy consumption. Therefore, based on these findings, for designing a probe climber, it is necessary to make further considerations on how to secure a proper frictional force and reduce slips between the drive wheel and the wire rope without increasing the pushing force between the wheels and the rope.

Roller material	Hardness
Polychloroprene (CR)	70–90
Nitrile rubber (NBR)	
Polyurethane rubber (UR)	

Table 9.2. Rubber materials for friction wheel

Then, impact on power consumption was verified when the pushing force of the idler wheel was changed. Figure 9.5 shows the case of the screw pushing mechanism, where the time history of power consumption, measured during elevation with the pushing force changed in three patterns: 68.2 N, 85.6 N and 97.5 N, indicates that power consumption during elevation

increased in accordance with the pushing force increase. A possible reason for unstable power consumption shown in Figure 9.5 is the effect of the wire rope vibrated by the ropes unevenness and the wind.

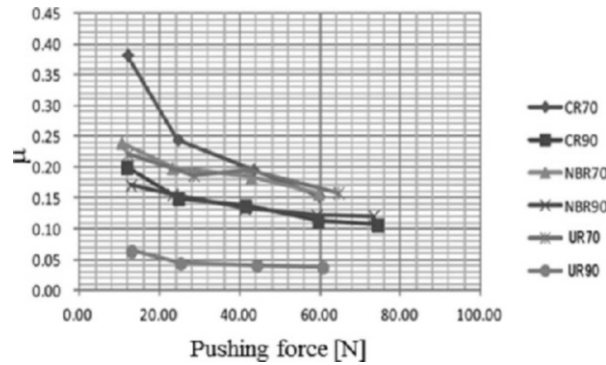


Figure 9.4. Change of the friction coefficient with the pushing force

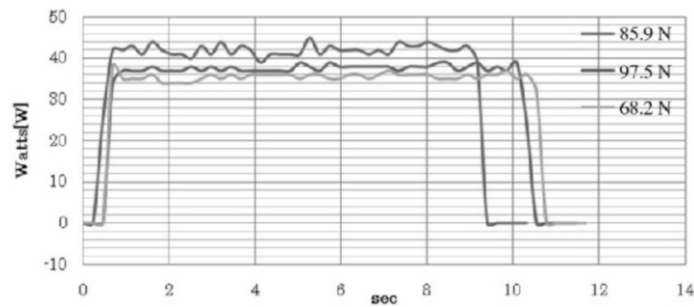


Figure 9.5. Pushing force and Power consumption

Based on the above measurement results, an improved probe climber was fabricated and shown in Figure 9.6 and the basic performance in Table 9.3. The improved probe climber was cable of stable elevation, because a bilateral symmetry layout was thoroughly ensured with multiple drive mechanisms and consideration was given to setting the center of the climber's gravity near the rope. Along with these, multiple drive wheels were used in order that the overall drive mechanism could reduce slips with the rope and vibrations. Also for the corresponding idler drive, the cross-section of the rope contact area was provided with a recess made of chloroprene material. In addition, the idler pulley shown in Figure 9.6, whose ball screw mechanism enables

position adjustment, makes it possible to adjust the contact area with the drive mechanism, depending on the wire rope tension. Furthermore, the climber is equipped with the function of regenerating potential energy by incorporating a power generating motor in place of a braking device used while an elevator lowers. Besides, as a countermeasure against regenerative mechanism failures, a ratchet mechanism shown in Figure 9.7 is also incorporated in order to prevent falling.

Additionally, an Xbee module and a directional antenna were used for communicating with the probe climber and it was verified that data communication with more than a 600 m distance was available. For the actual elevation experiments, in addition to monitoring the motor current, the voltage, the number of revolutions, a temperature sensor, an atmospheric pressure sensor, a humidity sensor, an acceleration sensor and a charge coupled devices (CCD) camera shown in Figure 9.8 were mounted.

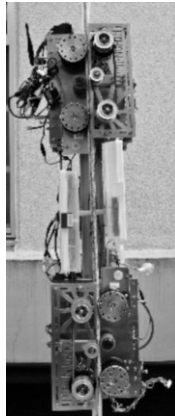


Figure 9.6. *Refined probe climber*

Length	1.2 (m)
Weight	9.5 (kg)
Number of motor	2
Maximum power of motor	2,900 (W)
Rotational Speed	247 (rpm/V)
Number of drive wheel	4
Maximum velocity	14 (m/s)

Table 9.3. *Technical specifications of refined probe climber*



Figure 9.7. *Ratchet brake*

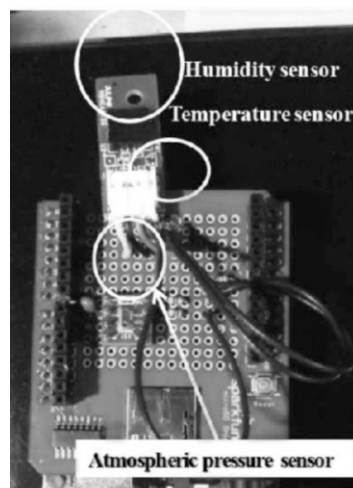


Figure 9.8. *Environmental sensor*

9.4. Damage detection test of elevator wire rope

Steel wire rope used for an elevator comprises of congregated twisted wires (called strands) made into a plurality of congregated twisted strands, with the fiber core as the axis. This is called a wire rope with n strands. Typically, 6–8 n strands are used for an elevator. In the damage detection test, as shown on the right side of Figure 9.10(b), 24 wires with 0.6 mm diameter was made into one strand. Wire rope made of six twisted strands was used, assuming the specimen wire rope with a cut in it (Figure 9.9(a))

have become loose and making use of two damage specimens in which wires as short as 10 mm were embedded (Figure 9.9(b)).

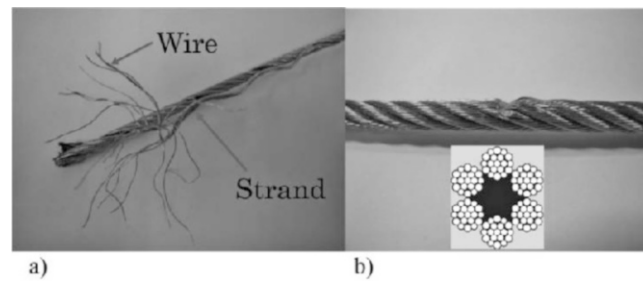


Figure 9.9. Damaged steel wire rope

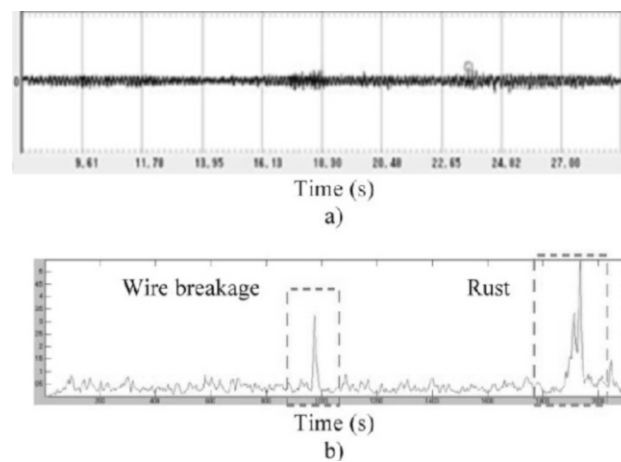


Figure 9.10. Time history of a) the response vibration and b) the leakage flux

First, the probe climber was used to take measurements with a wire rope tester for variations of response acceleration caused by damage to the wire rope and MFL caused by scratches and rust. Signal processing was performed on such measurement data to check the presence/absence of an abnormality judgment. The two signal processing methods were used: discrete Fourier transform (DFT) was used to analyze the power spectrum of each frequency and wavelet transform to express waveforms by scaling and shifting short waveforms called mother wavelets. Figure 9.10(a) shows the measurement results of response acceleration and Figure 9.10(b) shows that of MFL

strength. From the response acceleration measurement data (the original waveform), it is difficult to directly detect wire breakages and loosening, but from the MFL strength data, a clear change in the spectrum can be observed even in areas where wire breakages and rust have occurred.

9.5. Damage detection through signal processing

With the original waveform of the response acceleration in Figure 9.10(a), it is difficult to identify the presence or absence of damage and the location of damage. Thus, the two types of signal processing techniques were used to investigate the possibility of wire rope damage detection through response acceleration. First, frequency analysis was performed by applying the DTF to the original waveform of the measured response acceleration in the time period of confirmed variation of acceleration when the probe climber passed through damage area. Then, the specimens whose damage depths are 0, 0.9, 1.5 and 1.8, respectively, were compiled in the histogram shown in Figure 9.11, up to 20 Hz measured as the primary frequency band of response vibration.

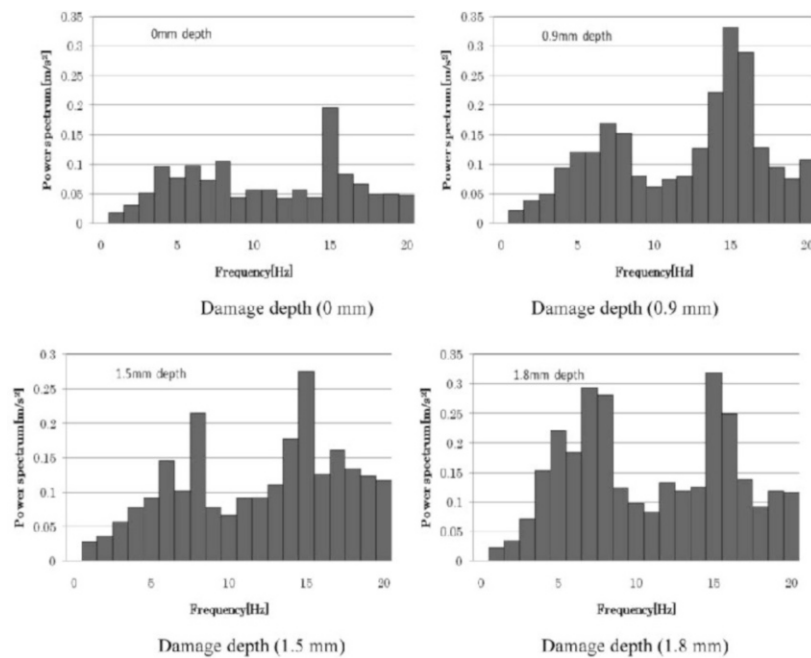


Figure 9.11. Frequency response property

In Figure 9.11, in each frequency up to 20 Hz, marked acceleration change caused by damage is confirmed in 5–8 Hz. In addition to this, large-scale power spectrum can be seen in the vicinity of 15 Hz, which, however, shows less change caused by damage depths. When compared with motor vibration data alone, it was confirmed that the power spectrum components came from vibrations of the probe climber's drive motor. Therefore, damage can be detected if attention is paid to acceleration changes in 5–8 Hz.

Next, in order to locate the occurrence of damage, time resolutions for measured signals are also important. Hence, signal processing was performed, using the wavelet transform that can ensure accuracy in both frequency resolution and time resolution. Figure 9.12 provides a comparison by the presence or absence of damage, regarding the results of the wavelet transform in the 4–8 Hz band, where the marked change was confirmed by the DFT applied to the response acceleration. It can be seen that the spectral intensity is remarkably greater in the right circle area in the lower part of Figure 9.13. When wire breakages were present above a certain level, response acceleration exceeding 0.2 m/s^2 could be detected in any of them. However, this integrity evaluation can be used only in the case of uniform motion. Even during the acceleration/deceleration of the automatic measuring device, response acceleration exceeds the threshold value as shown in the left circle in the lower part of Figure 9.12. This presents a possibility that response vibration caused by damage will be buried by the effects of acceleration/deceleration of the climber.

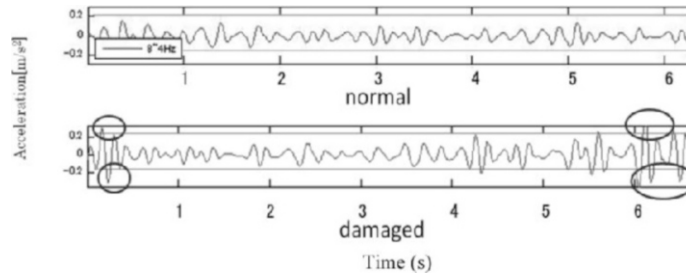


Figure 9.12. Wavelet transeform signal for the response vibration

9.6. Integrity evaluation of wire rope through MFL strength

Currently, an MFL measuring device of induction coil type is mainly used to detect wire breakages and rust on steel wire rope surfaces. As shown on the right side of Figure 9.13, this is a sensor system to detect small MFL

occurring in the areas of wire breakages and corrosion after a magnetizer organizes the magnetic lines of the wire rope, enabling checking of clear variations in MFL strength where slight damage or rust has occurred as shown in Figure 9.10.

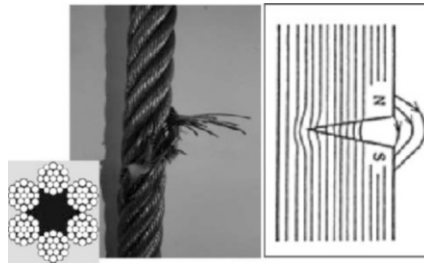


Figure 9.13. Breakage of rope strand and leakage flux

Figure 9.14 shows the result of wavelet transform applied to the MFL measurement data. Spectral variations can be also confirmed in the original waveform to the extent that damage can be detected. Then, Figure 9.15 provides the measurement results of the wire rope MFL strength in the number of wire breakages (0–5) equivalent to the initial damage, in order to examine the relationship between the MFL strength and the number of wire breakages.

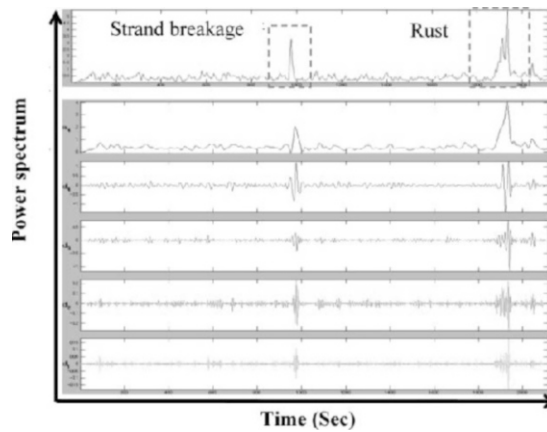


Figure 9.14. Wavelet transform signal of the leakage flux

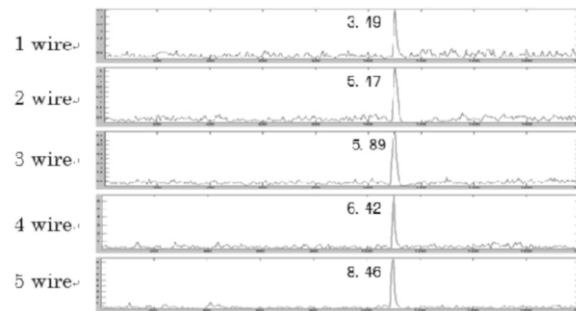


Figure 9.15. Leakage flux for the number of strand breakages

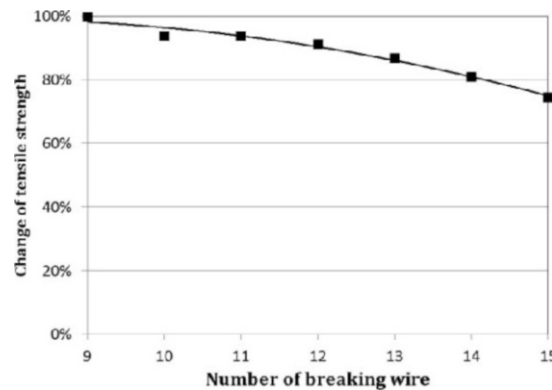


Figure 9.16. Effects of the tensile strength to the number of breaking wires

The results indicate that the MFL strength increases in proportion to the number of broken wires, which raises the possibility of estimating the number of initial external wire breakages. Subsequently, in order to verify the relationship between MFL strength and wire rope strength, a tensile test was conducted for wire rope with wire breakages and changes in tensile breaking strength were verified. Figure 9.17 shows the relationship between the number of wire breakages and the wire rope breaking strength in one strand of the wire rope (24 wires). In the range of the initial damage (about one–five wire breakages), there was virtually no decrease in the tensile strength. However, as the number of wire breakages approached 10, the tensile strength gradually decreased and it was observed that the wire rope strength was reduced by 10% or so, when 12 wires broke. Furthermore, in 12–15

breakages, a roughly linearly proportional decrease in the tensile strength was shown and 20% decrease in strength was observed in 14 wire breakages.

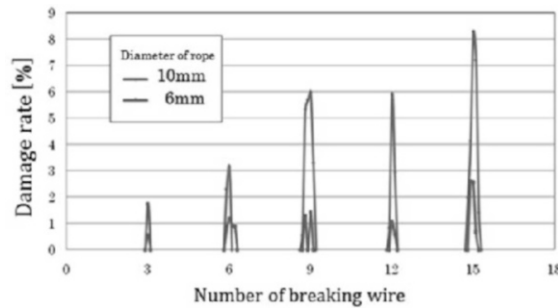


Figure 9.17. Effects of damage rate to the diameter of wire rope and the number of breaking wires

From the above discussion, an MFL measuring device of an induction coil type is an effective sensor system for detecting wire breakages, corrosion (rust) and wear on a wire rope surface. However, when it comes to handling the device, close attention is required to mount the device on the probe climber, as the device has a scanning speed limitation and is susceptible to the effects of vibrations. In addition, as shown in Figure 9.17, the device also has many demerits at present, such as decreased sensitivity in detecting small diameter wire rope and low accuracy in detecting inside wire breakages, such as in wire rope trough and bottom areas. Therefore, a sensor system was re-examined, which is light and compact, mountable on the probe climber and highly accurate in detecting damage. As an MFL detection method that has superior detection sensitivity for various forms of wire rope damage, a method using small Hall elements [SHA 12] can be considered, in addition to the induction coil type.

Hall elements were used because they have been conventionally considered difficult to use due to the temperature dependability but their recent developments have included III-V compound semiconductor InAs, which is highly sensitive and less temperature-dependent. As shown in Figure 9.18, a sensor system was built as a phased-array type that is reported to also enhance detection sensitivity. The phased-array Hall elements sensor prototyped in this study identifies relative damage positions and damage morphology from differences in respective response signals generated by a plurality of Hall elements arranged around the vicinity of the wire rope. This sensor, small and light, is also relatively easy to install into the probe climber.

The experiment conducted a comparative analysis of the cases of four and six Hall elements, respectively, and verified the effectiveness. Figure 9.19 shows the specimen onto which intensive wire breakages were provided on the wire rope surface. When the probe climber passed through this damage area, clear voltage changes could be detected in the Hall sensor system as shown in 9.19(b). From the above, the probe climber built in this study can detect damage and rust of wire rope, while elevating itself automatically. Its application to the automatic maintenance inspection of wire rope can be expected.

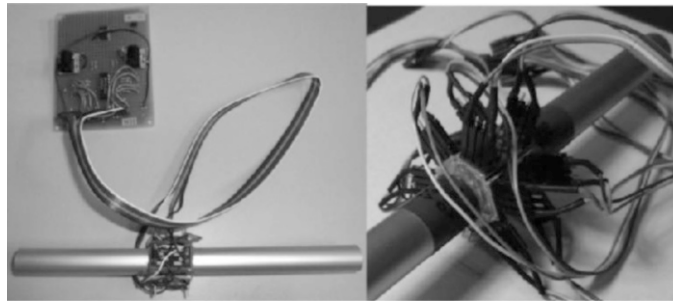


Figure 9.18. *Phased array hall sensor system*

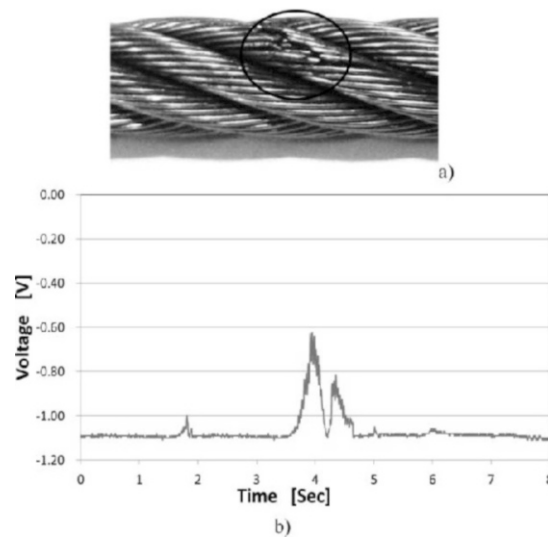


Figure 9.19. *Response signal of the phased-array Hall sensor system*

9.7. Damage detection of wire rope using neural networks

In the initial damage detection experiment for steel wire rope, the possibility was observed that MFL strength measured by the wire rope sensor could detect not only initial wire breakages but also rust occurrence areas. Then, damage detection using neural networks was tested in order to examine the possibility of automatic recognition by making the case analysis of wire breakages equivalent to the initial damage and rust occurrence. Table 9.4 indicates the output values by given measurement data and the estimated number of breakages after multilayer neural networks performed machine learning of 40 pieces of data on MFL strength (power spectrum) from initial wire breakages (zero–five breakages). The neural networks used for the machine learning were of a basic type with 10 hidden neurons, which has a two-layer structure that outputs to the output layer the case analysis of wire breakages and rust occurrence as well as the number of wire breakages, when an input is made to the input layer on the amount of variation and the spread of variation of the MFL strength: these networks after the postlearning neural networks show that they can almost exactly estimate the number of the wire breakages.

Input data of maximum power spectrum at the damage locations	Estimated power spectrum at the damage locations	Estimated number of breaking wire
0.98	0.98	0
3.49	3.30	1
4.82	4.97	2
6.39	6.27	3
6.46	8.26	4
8.39	9.04	5

Table 9.4. *Damage detection for the breaking wire of steel wire rope by the neural network*

9.8. Conclusion

The developed probe climber, an automatic-driving type inspection device for automatically detecting wire breakages and rust occurrence on wire rope, was prototyped and its performance was evaluated. The improved probe climber that was fabricated demonstrated that it could safely operate during repeated outdoor elevation of more than 500 m. In addition, a sensor system of phased-array type Hall elements was fabricated. Using the MFL variations, the system detected the locations of wire breakages and rust occurrence in the

initial stage and it was verified that the use of neural networks enabled distinguishing between wire breakages and rust occurrence by their distinctive signals and estimating the number of the wire breakages.

9.9. Bibliography

- [HAB 06] HABIB M.K., “Mechatronics engineering the evolution, the needs and the challenges”, *Proceedings of the 32nd Annual Conference of IEEE Industrial Electronics Society (IECON 2006)*, pp. 4510–4515, 2006.
- [HAB 07] HABIB M.K., “Real time mapping and dynamic navigation for mobile robots”, *International Journal of Advanced Robotic Systems*, vol. 4, no. 3, pp. 323–338, 2007.
- [HAB 08] HABIB M.K., “Interdisciplinary mechatronics: problem solving, creative thinking and concurrent design synergy”, *International Journal of Mechatronics and Manufacturing Systems*, IEEE Industrial Electronics Society, vol. 1, no. 1, pp. 264–269, 2008.
- [HAR 12] HARA E. *et al.*, “Application to the cable maintenance system of the space elevator climber”, *The Japan Society of Mechanical Engineers, Robotics and Mechatronics Conference 2012*, 1A2-L04, Japan, 28 May 2012.

Chapter 10

Fundamentals on the Use of Shape Memory Alloys in Soft Robotics

After 70 years since the origin of robotics, actuators still represent the real bottleneck for several robotic applications. The research in the field produced several technological solutions that can satisfy specific requests, but often constrains the designer into trade-offs to release other requirements that cannot be met. This phenomenon is particularly limiting when a mechatronics approach is considered since this concentrates on achieving optimal functional synergies among the system components from the earliest conceptual stages of the design process [HAB 07]. But if the design process is driven by the obviation of the actuation limitation, optimal performances cannot be reached. In this framework, a newborn branch of robotics which is now known as “soft robotics” deeply conveys this effect. Due to its nature and principles, the importance of a mechatronic approach to the design of soft robots will be underlined several times in this chapter. Its main peculiarity is to present the capability of a soft interaction with the environment and with the user while preserving robustness, reliability and the ability of powerful active actions. It is worth mentioning that there are at least two ways to obtain a soft interaction: the stiffness control of the robot actuators that act on the compliance of the robot joints [ALB 08] and the smart use of passive soft structural materials that assure an intrinsically soft impact [TRI 08]. Even if the two approaches aim at addressing the same result, the difference is much more than a sophism: in the first case, the impedance of the actuators is

controlled to provide low resistance during interaction and a stiff behavior when required; in the second case, the robots are almost totally composed of soft materials thus presenting compliance naturally, but they can change the stiffness of part of their body when necessary by generating the right combination of internal stresses. In this chapter, this second view will be considered and a particular, but already widely used, actuation technology for this kind of robot will be treated: the shape memory alloys (SMAs). The main characteristics will be introduced by placing a particular emphasis on the issues related to their use in soft robots.

10.1. Introduction

When talking about SMAs we should consider a wide family of alloys that show two main peculiar characteristics, detailed in the following section: the shape memory effect (SME) and the superelastic effect (SE). Since 1932, when a Swedish physicist Arne Olander discovered the SME on an AuCd alloy, a growing interest in this peculiar behavior led to the production of a huge number of different alloys presenting the same effects. These include certain copper alloys (CuAlZn), nickel–titanium-based alloys and some ternary alloys such as NiTiCu and NiTiNb. The most used and commonly available alloys are the NiTi-based alloys, which present the best memory and superelasticity. Discovered in 1961 by a group of US Naval Ordnance Laboratory researchers led by William Beuhler, the first commercial application is dated 1970, when Raychem Corporation’s “shrink-to-fit” pipe couplers were used to join hydraulic tubes in the F-14 Tomcat fighter aircraft.

For their own characteristics, SMAs are very well known in soft robotics and often taken into consideration when facing the actuation strategy to be chosen. The major advantage of this technology is its very high work density: the lighter the actuators, the bigger is the resulting power/weight ratio, therefore making SMAs a valid alternative to conventional actuators. Moreover, they are silent during operation and clean; they present high remotability and can be easily distributed to obtain local deformations with low driving voltages; they resist excellently to chemical corrosion and are generally biocompatible for their use in the medical field. On the other hand, some drawbacks limit the use of SMAs for specific applications: even if low driving voltage can be maintained, a relative high current is often required and it does not come with high energy efficiency (1–10%). Other disadvantages are a bandwidth limited by thermal cycles and possible arising of mechanical fatigue after a few cycles. Finally, the nonlinearity and the

wide hysteresis that characterize the thermomechanical behavior of the alloy generate several control issues. In fact, even if the SME of these alloys is quite trivial to observe, its correct and advanced use requires a deep knowledge of the intrinsic driving mechanisms.

Despite this, the interest on this technology still remains very high for its potential use as artificial muscles and it is proved by the tremendous number of past and ongoing studies conducted with the aim of improving usability and controllability.

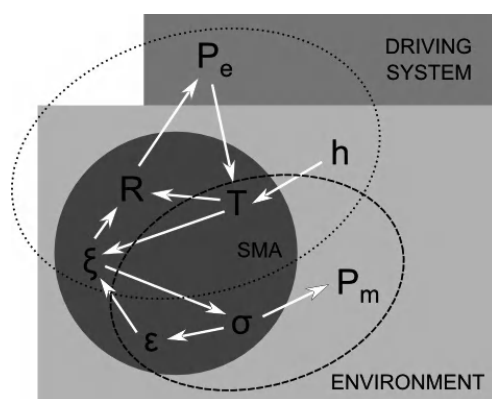


Figure 10.1. Schematic representation of an SMA-based actuation system: in the three main areas the variables that influence the behavior of the alloy and their dependencies are depicted: the temperature, T , the electrical resistance, R , the deformation, ε , the stress, σ , and the martensitic fraction, ξ , inside the alloy; the convection heat transfer coefficient, h , and the mechanical power acting on the environment, P_m ; the electrical power provided by the driving system, P_e .

In Figure 10.1, a schematic reconstruction of an SMA-based actuation system is reported. The gray filled circle represents the alloy encircled by the environment that is in close relation with it, while above all the driving system is responsible of actuation triggering. Inside every area the principal actors and their interactions are reported. Even if at this stage the complexity of the system could confuse the reader, it results are useful once every aspect of the system is explained in this chapter. After a phenomenological introduction (section 10.2), the thermomechanical behavior (section 10.3) and different modeling approaches (section 10.4) to describe the SME are illustrated. These involve the parameters underlined by the dashed circle in Figure 10.1 while some hints on their experimental measurement are provided in section 10.5. This first part of the chapter is more generally on

SMA actuators design principles are discussed in section 10.6, aiding the choice of the mechanical design needed for specific applications, followed by an overview on the most common fabrication methods (section 10.7). In section 10.8, the dependency among the variables underlined by the dotted circle in Figure 10.1 is clarified together with the explanation of the most used control methods. The topics illustrated in this second part can still be used for general purposes, but they are more focused on the use of SMA technology in soft robotics. The concluding section shows practical examples of successful coupling of soft materials with SMAs (section 10.9).

10.2. Shape memory effect and superelastic effect

The SME refers to the recovery of shape (i.e. strain) after apparent permanent deformation (induced at relatively cold temperatures) by heating above a characteristic transformation temperature (TT). Depending on the applied thermal treatment, when the temperature drops below TT the spring remains in the recovered shape if the alloy presents the one-way memory effect (OWE) (Figure 10.2(a)) or returns to the deformed shape if it exploits the two-way memory effect (TWE) (Figure 10.2(b)).

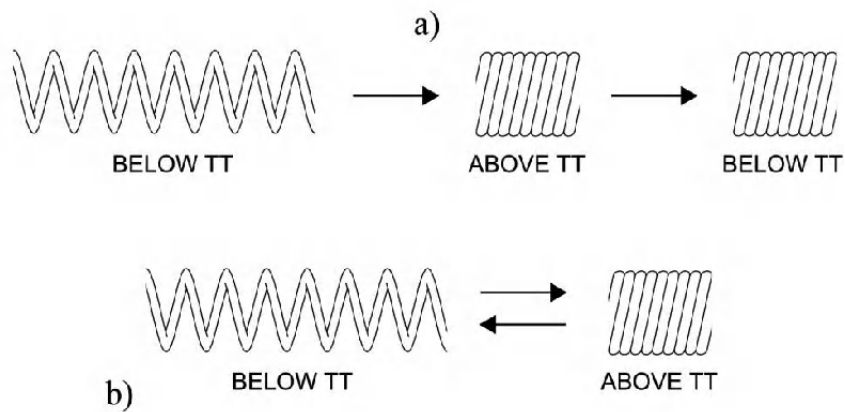


Figure 10.2. Spring showing the SME: after apparent permanent deformation induced below the TT the alloy can recover the initial shape by heating above the TT; after temperature drop a) OWE or b) TWE can be observed

SE refers to the isothermal recovery of relatively large strains during a mechanical load–unload cycle that occurs at temperatures above TT (Figure 10.3).

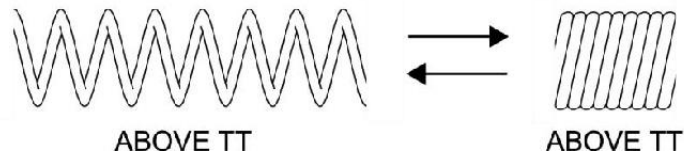


Figure 10.3. Spring showing the SE: the alloy is always above the TT , thus every applied deformation is immediately recovered once the load is removed

Even if the distinction between SME and SE is reported in every text introducing SMAs, it is worth remembering that the two effects are the phenomenological demonstration of the same behavior. In section 10.3, the two effects are thoroughly analyzed from a thermomechanical point of view and this distinction will be clarified.

10.3. SMA thermomechanical behavior

SMAs owe their name to the capability of recovering a predetermined geometrical shape (memorized), after a plastic deformation, by a thermally activated martensitic transformation. This effect is based on the presence of two stable crystalline phases: martensite (or α -phase), which is thermodynamically stable at lower temperature, and austenite (or β -phase), which is the parent phase stable at higher temperature. Martensite presents a crystal structure with low symmetry, such as tetragonal, rhombohedral, orthorhombic, monoclinic or triclinic depending on the composition of the alloy, while austenite has a higher symmetry usually based on a cubic lattice (Figure 10.4).

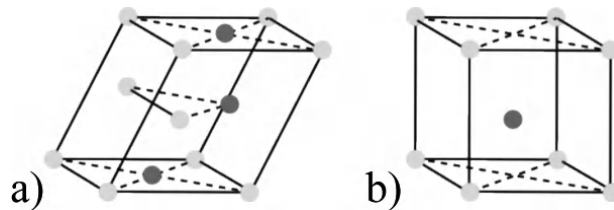


Figure 10.4. Crystal structure of a) martensite and b) austenite

For the sake of completeness, a third meta-phase should be mentioned. Between martensite and austenite, the so-called R-phase sometimes appears, but its presence deeply depends on the composition of the alloy and since the strain amount developed during R-phase is insignificant as compared to the strain amount developed during martensite process, it is very often neglected.

The fundamental difference between martensite and austenite crystal structure is also the basis of the SME. The transformation from austenite to martensite needs a rearrangement of the alloy microstructure leading to oriented martensite (detwinned, stress-induced) or thermal martensite (twinned, self-induced).

Thermal martensitic transformation is reversible and diffusion less: neighboring atoms in the lattice will remain in equal relative positions after the thermally driven transformation, even though the relative distance will be altered and the passage from one phase to the other is always possible and only determined by the thermal conditions applied. When the temperature rises above the reverse transformation start temperature (A_s), martensite starts to transform into austenite and the process is complete when the reverse transformation finish temperature (A_f) is exceeded and the alloy is fully austenitic. In the same way, when the alloy is cooled down below the martensite start temperature (M_s) the austenite starts to transform into martensite until the martensite finish temperature (M_f) is achieved (Figure 10.5). A common way to describe the state of the alloy in a certain moment is using the martensite fraction (ξ) that indicates the percentage of structure in martensitic phase. By adding a mechanical load, the characteristic TTs change introducing the stress dependency (stress gradient) is shown in Figure 10.5.

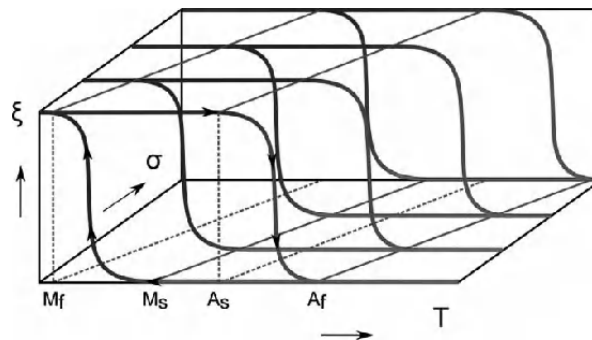


Figure 10.5. Diagram showing martensite fraction, ξ , versus temperature, T , relationship at constant loads, σ . Thermodynamic cycles are shifted rightwards (stress gradient) increasing the transformation temperature values

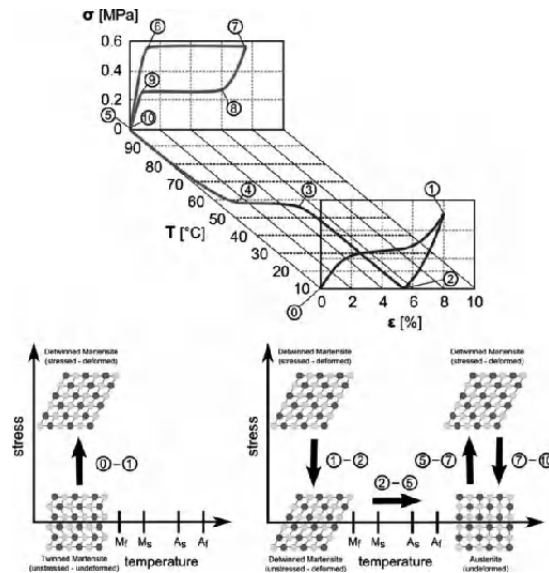


Figure 10.6. Thermomechanical behavior of an SMA showing the dependency among temperature, T , stress, σ , and strain, ϵ , and the related crystal structures

Other than changing the characteristic TT values, if a mechanical load is provided during the thermal martensitic transformation cycle, a more complex behavior appears. Temperature, stress and strain are correlated following the relation shown in Figure 10.6 that also highlights the SME of such alloys. In point (0), the material is maintained at low temperature in an undeformed and unstressed state presenting a randomly organized twinned martensite. By applying a stress, a deformation is induced showing a nonlinear response (1) and if the deformation is maintained below 8–10% the only structural change that occurs is a detwinning process that tends to orient the martensite domains. The return to an unstressed condition is associated with a large hysteresis and a residual strain (2) since the reorientation occurred in point (1) is caused by an apparent plastic deformation. On increasing the temperature, a thermal cycle is commenced and as already described the alloy undergoes a structural transformation. In (3), the A_s is reached and martensite starts moving into austenite, while in (4) the process is completed. By continually increasing the temperature, no other effects are obtained (5). This concludes the description of a cycle where the SME is shown. If from this point a load is applied again, a different behavior is shown this time: the higher temperature maintained the alloy in austenitic phase so that the response is much stiffer and linear (6). A higher stress leads

to the appearance of a plateau that is justified by a stress-induced microstructural change: austenite is transformed into detwinned martensite (7). But since martensite is not stable at high temperatures when the load is removed, the reverse transformation occurs (8) until the alloy is again fully in austenitic phase (9). Point (10) concludes the transformation cycle at high temperature, thus describing the SE effect.

10.4. SMA constitutive models

As shown in the previous section, the SMAs exhibit a nonlinear and pseudoelastic response that varies depending on the temperature and loading conditions. An accurate model of this behavior is a necessary prerequisite for the use of the SMAs as actuators. A mathematical expression in a form that is amenable to incorporation into other engineering tools (finite element procedures or control analysis programs) would enable a fast design of SMA-based actuators. Unfortunately due to the complex interdependencies among the variables controlling the SMAs behavior, these materials have been the subjects of numerous constitutive models, without a real supremacy on using a particular one. Several models have been proposed, everyone following a different approach. In the classification Paiva *et al.* divided and grouped the most valuable works under the five kind of models listed in the following:

- 1) models that take into consideration the microscopic (molecular level) or mesoscopic (lattice particles) phenomena;
- 2) models based on a macroscopic approach (phenomenological features);
- 3) models with assumed phase transformation kinetics;
- 4) models based on elastoplasticity theory;
- 5) models based on plasticity concepts.

The second group includes the simplest models and they are often used when an in-depth description of the microscopic phenomena is not necessary. From an engineering point of view, this could sometimes be too much limiting and, for these reasons, the most popular approach is described by the third group where pre-established simple mathematical functions are used to describe the phase transformation kinetics, thus avoiding a too complex mathematical approach.

Of course, also in this case, some assumptions are necessary and even in the luckiest eventuality of a level of description that meets all the designer

specifications, for the design of a specific actuation system, some thermomechanical tests are required anyway. This could appear as a limitation, but from the perspective of a controls engineer, any model that can capture the SMAs response adequately and efficiently (possibly in real time) is a “good” model, irrespective of whether that model has been built on physical principles or purely empirically.

10.5. Hints on SMA thermomechanical testing

Since the testing of SMAs is not yet standardized and unlike conventional alloys, material property tables either are not available or provide incomplete, or even incorrect, information for the user. Commercially available SMAs can present a wide range of possible combination of performances, thus users are often forced to undertake testing SMAs in their own laboratory to obtain a satisfactory characterization of the material at hand. Whatever the model chosen to describe the alloy behavior, in the design phase there are at least three fundamental data whose precision can greatly affect the prediction of the results: the characteristic TTs (A_s , A_f , M_s , M_f), the Young’s modulus of both austenite and martensite phases and the stress gradient.

Differential scanning calorimetry (DSC) is the most precise technique to obtain a complete description of the thermal properties of the alloy. An electrical resistivity scan is a simple alternative method to identify the characteristic TTs, especially if a DSC machine is not available, but results are difficult to interpret and it provides limited data on latent heat and specific heat.

Once DSC thermograms or electrical resistivity scans have been performed to establish the relevant TTs, in order to assess the remaining data, it is necessary to obtain an overview of the material behavior with respect to axial load–elongation–temperature space (schematically reported in Figure 10.1 to show the common trend). This is usually observed by performing isothermal and isotonic tests using a tensile machine with a thermally controlled chamber: while imposing a constant temperature, elongation is swept up and down and the load response is measured; while imposing a certain force, temperature is swept up and down and the strain response is measured. One possibility to assess the deformation is measuring the global strain directly from the displacement of the testing machine’s crosshead, but it does not take into account possible local effects due to a non-uniform deformation of the material. Another possibility is to measure local strain (namely only the gage section of the test specimen) through mechanical or optical systems. In both cases, special precautions should be

considered in gripping the sample to avoid slippage and stress concentration at the ends. Moreover, to obtain correct results, the strain control imposed to the machine should be based on global deformation (not local) and on a deformation rate of approximately $\dot{\delta}/L = 5 \times 10^{-5} \text{ s}^{-1}$. During these tests, a very precise temperature control is also mandatory: Joule effect heating is one of the most used for its simple implementation, but environment conditions and power supplied must be very well controlled; environmental chambers and a liquid bath can be useful to cover a broad range of temperature but infrared measures could be disturbed; conduction contact could be exploited using a temperature-controlled heat sink in thermal contact with the specimen.

For reference purpose, typical values that describe thermomechanical (but also chemical and magnetic) properties of a NiTi alloy are presented in Table 10.1.

Melting point	1310°C
Density	6,450 kg/m ³
Thermal conductivity, austenite (martensite)	18 (8.6) W/m°C
Thermal hysteresis	15–30°C
Specific heat	470–720 J/kg°C
Transformation enthalpy	3.2–12 kJ/kg
Transformation temperatures (unstressed)	A_s 75°C A_f 88°C M_s 68°C M_f 60°C
Stress gradient	0.12°C/MPa
Electrical resistivity, austenite (martensite)	100 (80) μΩcm
Magnetic permittivity	1.002
Maximum deformation (OWE)	8%
Maximum deformation (TWE)	2%
Young's modulus, austenite (martensite)	83 (28–41) GPa
Yield stress, austenite (martensite)	200–800 (150–300) MPa
Breaking load (cold worked)	1500 MPa
Fatigue limit (10 ⁶ cycles)	350 MPa
Corrosion resistance	Excellent

Table 10.1. Typical values for a NiTi alloy

10.6. Design principles

Once the thermomechanical behavior of the alloy has been assessed, the design of the actuation system cannot continue without considering its embodiment. Although the introduced concepts still have a general connotation and can be applied in any application, for the remainder of the chapter, in several cases the considerations have a narrower applicability focusing on SMA-based actuation systems for soft robotics.

When the design phase of the actuation system begins, there are some important aspects that should be taken into consideration for a correct setup of the device.

10.6.1. Geometrical choice

The most important factor in the design phase is represented by the definition of the geometrical parameters of the alloy to be used. As discussed earlier, SMAs show very unique mechanical performances in terms of power density, but very limited deformation capability. Thus, while theoretically SMA wires could be used as the simplest linear actuators ever seen, its use in straight configuration is suitable for a very limited number of applications. To meet specific requirements, SMAs are more often used as an active material to be shaped in the right configuration with a trade-off between available force and deformation capability.

SMAs are commercially available in different forms: wires with different diameter and cross-sections, ribbons with different geometrical proportions, thin films and plates for sputtering deposition. The most important driving factor on the choice of the right design of the actuation system is represented by the description of the tensional state of the material during its working cycles. In general, a material can be subjected to two kinds of stresses: normal and shear stress whose combination determines three different deformation states: tensile, bending and torsion. Each of these deformations can be exploited as strained initial configuration for the alloy, but they present different efficiency depending on the average stress imposed across the section. In the optimal case, all the material is subjected to the same stress so if the maximum allowed stress is not exceeded, all the recoverable deformation is uniformly used. Any tensional state different from this will exploit only partially the capability of the alloy.

1) Tensile deformation is obtained when an axial load is applied to a body (e.g. a bar subjected to pure tension or compression), so that the force passes

through its centre (Figure 10.7, left). In this case, the stress, σ , can be obtained by dividing the total normal force, F , by the bar's cross-sectional area, A . In this case, the tensional state of the body is homogeneous since all the material experiences the same stress level so representing the optimal way to exploit all the potentiality of the material.

2) Bending deformation generates compressive and tensile stresses at any cross-section that vary linearly (Figure 10.7, center). They depend on the applied bending moment, M_b , on the distance of it from the neutral plane, x , and on the moment of inertia, I . In this case, the maximum strain will occur in the upper and lower part of the material, while all the intermediate states will present partial deformations that translate in a lower produced force. For example, if a rectangular cross-section is considered, given the same maximum allowed stress, the efficiency in using the recoverable capability under bending deformation is reduced to the half with respect to a tensile deformation.

3) Torsion deformation is obtained under the effect of pure shear stresses (Figure 10.7, right). Similarly to the previous case, it depends on the applied torsion moment, M_t , on the radial distance of it from the neutral axis, r , and on the polar moment of inertia, J . Also, in this case, there is a sensible reduction in the efficiency that can be calculated by fixing the cross-section geometry. If a circular wire is used, it is easy to demonstrate that a reduction to one-fourth with respect to the optimal situation is found.

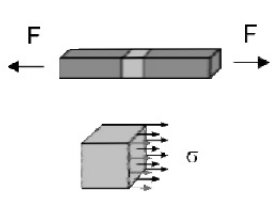
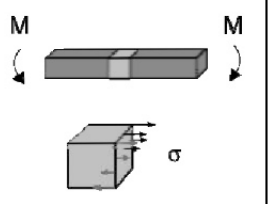
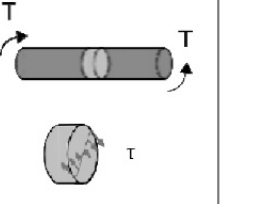
Tensile deformation	Bending deformation	Torsion deformation
		
$\sigma = F/A$	$\sigma(x) = M_b x / I$	$\tau(r) = M_t r / J$

Figure 10.7. Possible deformation states deriving from different load cases

Of course, the same principles apply for any material subjected to a stress, so that this analysis is still valid also in the eventuality of the presence of soft

materials in the actuation system. The difference is related only to the different characteristic stress–strain curve given by the nature of the soft material in use: elastomers, for example, are considered incompressible and usually show a hyperelastic behavior that consists of reaching very high deformations (also above 1000%) under relatively low stresses without presenting plasticity phenomena due to the random disposition of the polymeric chains.

As remarked introducing the thermomechanical characteristics of the alloys, excessive deformations are the most dangerous situations for maintaining the electromechanical performances of the actuator. In practice, the major risk is represented by the eventual presence of unknown inputs, not considered in the design phase of the mechanical stresses allowed. An excessive stress causes a relative shifting between crystalline planes that leads to permanent plastic deformation. Other than predicting the maximum applied stresses and avoiding undesirable deformations, factor of safety should be applied in order to stay well before the maximum allowed stress. The diagram shown in Figure 10.8 can be used to identify the most suitable maximum stress in relation with the maximum strain; depending on the maximum cycles, the actuator is designed to operate before fatigue starts negatively affecting the performances.

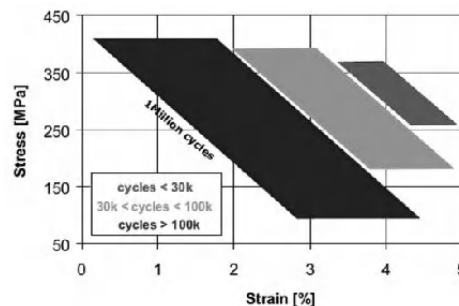


Figure 10.8. Diagram required to assess the stress–strain trade-off depending on the expected actuator life (Source: SAES Getters)

The design process of the actuator begins with the definition of the necessary force that the actuator should generate and from this value the correspondent internal stress can be calculated. The diagram reported in Figure 10.6 clearly shows an abrupt change in the slope of the curve (point 6) that represents the beginning of the stress-induced martensite process and its stress value depends on the temperature. Thus, it is evident that any

temperature above the minimum required temperature satisfies the force requisites, but there are particular advantages in choosing a temperature whose curve presents the elbow immediately after the target force: a well-addressed choice can lower the power consumption and can be used to exploit the SE to preserve the alloy while experiencing unexpected overelongations.

10.6.2. *One-way or two-way memory effect*

The thermomechanical behavior described in section 10.3 is due to the intrinsic properties of the SMAs. It demonstrates how the alloy can recover the initial configuration by increasing the temperature (OWE), but there is another option (already introduced in section 10.1), that is the use of the TWE. In this latter case, the alloy possesses two “memorized” states: one associated with the low temperature (in martensite phase) and one with the higher temperature (in austenite phase). This means that virtually there is no need of a bias system that restores the initial shape of the actuator.

At first, we can think that the use of the TWE is much better than the OWE, but there is a series of drawbacks that should be considered:

1) The TWE is an acquired property, meaning that a complex thermomechanical process is necessary including several shape memory cycles followed by superelastic cycles (more details can be found in [DUE 90]).

2) The recoverable strain that the alloy is able to resist before “losing its memory” is much more limited: it rarely exceeds 2% that in turn limits the produced force too.

Thus, the choice between OWE or TWE widely depends on the specific application, but also on the possibility to add a restoring system with the aim to provide (or at least support) the necessary force to return to the initial strain.

10.6.3. *Restoring force*

Connected to the previous paragraph, if an external restoring force is preferable, there exists three different ways of implementation:

1) *Constant force*: this is the simplest case where a suspended mass or a linear spring is connected to one end of the alloy and its force is used to

restore the actuator in the deformed configuration, in order to allow starting a new working cycle. From a design viewpoint, it is important to note that in this case there is no shift in the activation temperature values since the applied load is always constant.

2) *Variable passive force*: another alternative solution is represented by the use of a bias spring that provides a force linearly dependent on its deformation. Even if a few additional design efforts are necessary, this is the most used solution.

3) *Variable active force*: the most refined method to provide an antagonistic restoring force is to place a second SMA that acts to counterbalance the effect of the first SMA. The alternate activation of the two actuators allows the completion of the working cycles and virtually they could also be used to finely control the position of their connecting point by balancing the two generated forces.

The last two options cover nearly all the applications of SMAs in robotics and the same trend is proofed also in the narrower field of soft robotics, especially the second solution listed above. Typically, soft robots present a compliant and flexible body that already stores the elastic deformation energy produced by the actuators (whatever they are), thus providing a restoring force that can be used to return to the strained configuration. This represents the best integration of an actuation system which synergistically works with the robot body. On the other hand, this possibility can be pursued only if body material parameters can be adjusted without constraints. In this sense, the choice is not nearly trivial, since typically elastic properties of the body have to meet also other specifications and a trade-off is necessary. Moreover, the mechanical response of the elastomers is anything but linear, thus the mechanical coupling between the elastomer and the SMA response has to be taken into consideration. Fortunately, often tensile–compression tests (carried out setting similar working conditions, especially for the deformation range) are sufficient to derive the data to describe the behavior of the coupled system (Figure 10.9).

10.6.4. *Anchor points*

Another important point to be taken into consideration is a practical aspect. SMA actuators are often used as wires (straight or whatever shape) meaning that they need anchor points where the ends have to be fixed. These points have to guarantee mechanical stability and avoid a too local point of force application that (after several cycles) could lead to a shift of the terminals or to the total failure of the actuator. Moreover, these points are

often used as the most suitable place where electronic connections are lodged. In the simplest case (if the application allows it), the alloys can be mechanically fixed on a rigid plate (crimped in hollow cylinders or with bolts) that can be embedded as floating parts in an elastomeric matrix and providing a perfect electric connection site. Where the use of rigid plates is not allowed, a more complex solution has to be used. But this issue raises an important question. In soft robotics, hard materials should not be used, but “hard” is a relative concept. What is the boundary between hard and soft? Consequently, how can the right properties of the needed components be addressed? Usually, in soft robotics there is a huge use of elastomeric materials and the choice of the most suitable material is dictated by the overall passive properties that the robot is designed to present. Thus, one tentative answer to the previous question is: the boundary between hard and soft is traced by the overall passive properties of the robot so every embedded component or sum of components that dramatically change this passive behavior has to be considered too hard and consequently should be discarded.

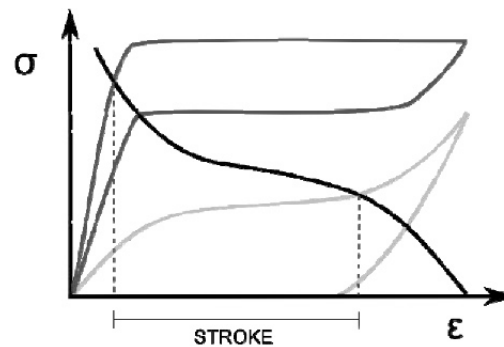


Figure 10.9. Stress versus strain relationship of an SMA in martensite (dark gray) and austenite (gray) phase that acts against an elastomer (light gray). The intersection points among the curves dictate the maximum available stroke that the system can implement

Possible connection sites can be easily introduced by using braided fibers (commonly used also to couple tendon-driven actuation systems and soft materials) available in different forms and materials. Woven or knitted textiles and braided sheaths can be professionally manufactured in order to meet specific dimensions and shapes and they can be easily embedded in a soft matrix. Of course, this inclusion affects the elastic properties of the soft material; so that it should be considered in the body of a global passive

response and once again shows the importance of a design approach that takes into account all the components of the system from the very early phases.

10.7. Fabrication methods

When the actuator specifications can be met using a straight geometry, wires or ribbons of the desired dimension and cross-section can be easily found off-the-shelf, but in most cases more complex geometries are necessary. There are several methods to obtain particular shapes; here four of the most common methods will be shown.

Manual fabrication of the SMA actuators can represent a fundamental early step in the design process to realize a large number of prototypes in a relatively fast way at low cost, before using more advanced technologies. Therefore, a good fabrication method would be preferable to obtain repeatability, since small differences and imprecision can greatly affect the mechanical performance of the actuators. The manual method starts from raw material like wires or ribbons that will be shaped in the final configuration through a heat treatment. To memorize the desired shape, the spring has to be mechanically constrained and maintained at 450°C for 30 min inside a furnace under flowing nitrogen or other inert gases (to prevent oxidation). After this time has elapsed, the alloy is water quenched.

Laser machining provides a design flexibility that is unmatched by other techniques. It allows the designer to simply alter a computer drawing in order to modify the final product and the feature sizes can be maintained on the order of 25 μm . This technique can occasionally cause heat damage, which manifests as a heat-affected zone (HAZ) at the cut edges, and can affect the performances of the actuators. In this case, postprocessing such as electropolishing on these actuators could be used to eliminate the HAZ and to smooth the surface characteristics.

Sputter deposition of NiTi films is a valid alternative to the use of foils [WOL 95] and it can be utilized for fabricating and patterning at the same time. With this method also, very thin layers (normally less than 10 μm in thickness) can be deposited on silicon, glass or polymeric substrates providing the first step of eventual other fabrication steps.

The wet chemical etch process has been also successfully tested and involving the use of soft lithography, it adds the possibility to obtain a batch

process where SMA foils are processed in compound with SU-8 [LEE 08]. It allows a precise fabrication of very small planar actuators and its contemporary embedding in a fixing component.

One last topic regarding the fabrication method is about joining SMA. Once the stable mechanical fixing is provided, the alloy has to be electrically connected to allow the current flow and its activation. In the case of a mechanical anchor point, the wire terminals can be simply crimped together with the alloy. Although the mechanical solution is considered the simplest and most reliable, there are other more elegant solutions to join SMA for electrical connection [WU 01]: welding an SMA to itself is possible with CO₂ laser, Nd:YAG laser, tungsten inert gas (TIG) and resistance welding without affecting the SME of the alloys; soldering is possible using halogen-based fluxes or mild fluxes if the surface is plated with Ni or noble metals; joining SMAs with other dissimilar metal is more challenging and needs the use of proper interlayer materials.

10.8. Activation methods and control design

The shape of an SMA is based on thermal control, but there exists at least two ways of causing a temperature increase: direct heat flow and Joule effect. While the first is usually used only in very particular cases where the alloys are used to monitor the temperature of an environment (fire alarms) or to control it in order to maintain always the same temperature (water valves), the second one is the most used for an active control of the alloy state and can be electrically or inductively driven. An SMA actuator can be thought as a variable resistance due to its geometrical and structural modifications. The main problem in controlling SMAs is the wide hysteresis shown during their thermal cycles (as already shown in Figure 10.5).

Another fundamental point to remember is that the systems based on thermal effect are deeply affected by the environmental conditions and neglecting thermal properties of surrounding materials could turn into performances worsening. Thus, in the case of soft robots where the SMA actuators are embedded in a liquid or solid/elastomeric means, part of the design of the control system is actually strongly connected to the choice of the right materials to be used for the robot itself.

Depending on the desired application, different activation methods can be used and the choice of the control types also depends on the geometrical properties of the actuation system and on the complexity of

the environment. The general equation that describes the power transformation and the relation among the variables that play a role in the thermal process is as follows:

$$\rho c V \frac{dT(t)}{dt} + \rho V \Delta H \frac{d\xi}{dt} = Ri^2(t) - hA(T(t) - T_a) \quad [10.1]$$

The first two terms represent, respectively, the power used through Joule heating to raise the actuator temperature, where (except for the more obvious meaning of the symbols used) ρ is the density of the material, c the specific heat, V the volume, H the latent heat for the transformation and ξ is the martensite fraction in the alloy at a given time; the third term is the electrical power delivered to the actuator; the fourth term quantifies the heat dissipated on the environment where h is the convective heat transfer coefficient and T_a is the room temperature.

The velocity of the heating process depends on the supplied electrical power, but the limiting factor on the working frequency is represented by the cooling phase. It is clear from equation 10.1 that the velocity of this phase can be increased in two different ways: increasing h by using the right geometrical conditions (see section 10.8.1) or maximizing the external area, A , which exchanges heat with the environment. Forced convection and Peltier effect can be used in this sense. When SMAs in straight wires or springs are used there is another common possibility to affect the heating-cooling cycles: to fit a sheath around the alloy. Of course, its mechanical contribution has to be negligible (or at least considered in the force balance), but also from the thermal point of view the critical radius notion has to be taken into consideration: basically, if a thermally non-conductive material is placed around the alloy, this additional insulation increases the conduction of the insulation layer, but contemporaneously decreases the convection resistance of the surface. It means that there is a radius that maximizes the heat flux and every bigger or smaller radius obstacles heat dispersion.

This is a general principle, but finds a special importance in soft robotics because sometimes the SMA actuators are directly embedded inside an elastomeric body thus heavily conditioning the thermal cycles. Polymers are usually thermal insulators and elastomers are not an exception. Of course, in the thermal conductivities list it behaves better than air, but it is an obstacle to heat dispersion with respect to other materials such as water or metals. In this case, liquid fillers could be used to speed up the cooling phase, but particular attention is needed to avoid the eventual appearance of chemical reactions induced electrically.

10.8.1. *On-off activation in open-loop control*

The simplest case one can imagine for an application of an SMA actuator has very well known conditions both for the thermal and mechanical properties of the system and for the external environment that will affect the actuator cycles. In this kind of application there is no modulation of the produced force, it only guarantees the repeatability of the same performances every time the actuator is activated (on-off activation). In this case, the dynamic transient of the heating process is not known, but for an on-off activation this is not necessary. This means that once fixed the mechanical parameter of the actuator (in order to meet the application), equation 10.1 reduces to the third and fourth terms and it can be used to derive the right electrical power needed to obtain the full activation of the material. This kind of approach implicates:

- 1) Numerically, the resistance of the alloy has to be set to the austenite phase value. This approximation leads to an underestimation of the initial resistance value, but it self-adjusts while heating up.
- 2) The driving system should be able to supply the same power whatever is the electrical load (in order to fix the value of the third term of equation 10.1), since the applied voltage and the drawn current have to be maintained insensible with respect to resistance variations.
- 3) The geometry has to be simple enough to allow an acceptable description of the cooling term through the expression of the parameter h that is often an experimental approximation. In [WEL 08], a wide collection of examples can be found for laminar and turbulent flows for forced convection as well as natural convection for several geometries.

If the geometry of the system presents a higher complexity, the thermal conditions evolution could be hard to predict. The known models to describe the heat exchange have been derived for simple geometries, but shifting toward more advanced shaping, such as helicoids or planar springs, they lose their accuracy. Moreover, when more independent actuators work in a limited space, a combined thermal action has to be considered: the activation of one actuator influences the environment that in turn can affect the performances of the nearby actuators' thermal cycles. In this case, the most effective solution is represented by the use of finite element analysis programs, where once virtually reconstructed the geometry of the system and a set all the thermal and electrical properties, a multiphysics analysis can aid the evaluation of the optimal working conditions as well as the most critical points and it allows the formalization of the fourth term of equation 10.1.

10.8.2. Modulated activation in closed-loop control

The proportional control of SMAs based on mathematical models often does not have sufficient accuracy and for this reason feedback control is very often utilized. This method is the most complex, but the most precise, even if it requires a more sophisticated design. In this case, the uncertainty of the environment and the complexity of the system can be increased as required since the state of the alloy is continuously monitored by sensors. There are two available ways to close the loop: external variables feedback or internal state variable feedback. The first family is based on the measurement of displacement (by using strain gages, mechanical extensometers or optical sensors in parallel configuration with respect to the actuator stroke direction) or force (with load cells). Unfortunately, the knowledge of these two variables is often not enough to assess the real state of the alloy since they cannot reveal temperature disturbances or external loads and as a result they are likely to cause overheating. On the contrary, monitoring the internal state of the alloy is more effective and it consists of measuring temperature and/or electrical resistance. Several methods of measuring the temperature could be applied such as thermocouples or non-contacting infrared temperature probes, but they present very limited usability for microscale applications. Nowadays the electrical resistance measure remains the most used and effective feedback to control SMA-based actuators. In fact, the measure of inner electrical resistance allows us to detect the alloy martensite fraction. The method consists of measuring the electrical resistance of an SMA element, calculating a maximum safe heating current as a function of measured resistance and ensuring that the actual heating current does not exceed this maximum value. The implementation of this control is based on a linearized model that correlates the strain versus the electrical resistance of the alloy that usually presents a fairly small hysteresis (Figure 10.10).

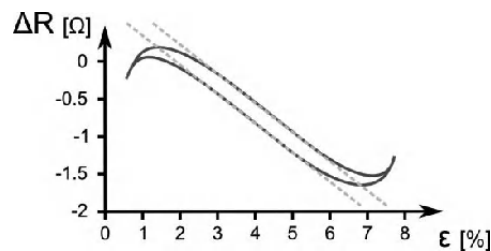


Figure 10.10. Resistance versus strain curve. Because of the low hysteresis, the linearized function can be used in the control loop

The only limit of this method is due to the sensitivity of the alloy to external loads. Antagonistic forces applied on the actuators introduce a new variable that make the relation non-univocal: one cannot know if the resistance change is due to an external stress that stretches the alloy or to a temperature change in the environment. In this case, the stress has to be measured together with the resistance, so that the right strain–resistance curve can be selected.

In the implementation of a closed-loop control besides a few attempts with elaborated strategies using complex hysteretic models to compensate for the material behavior [HAS 98] or neural networks and sliding-mode robust controllers for precision tracking of SMA [SON 03], the most cited and described in the literature are based on proportional, integral and derivative (PID) control even if they have been found to produce steady-state errors as well as overshooting and undershooting in heating and cooling phase, respectively. Both continuous control and modulation techniques are widely used, but pulsed driving methods have been proved more advantageous: pulse width modulated-PD controller, pulse width pulse frequency modulation-PD controller and the continuous PD controller have been compared under the same conditions demonstrating that at the cost of a reduced response time the performance of the pulse width (PW) modulator is comparable to that of a continuous controller in terms of control accuracy, but offers dramatic energy saving and improves the system robustness to external disturbances [MA 03].

10.9. Applications in Soft Robotics

Soft robotics is a newly born field that has now started showing, first, convincing results. In this framework, the coupling of soft and/or flexible materials with SMA actuators has been successfully implemented, especially in biomimetic and bioinspired robotics. Caterpillars, worms and octopus arms are the most cited and studied. Reproducing the capabilities of such animals requires, in fact, the presence of a soft body with active abilities of motion without the use of rigid structures.

Among several caterpillar-like robots, one of the most effective examples is represented by the GoQBot [LIN 11]. It replicates a very fast self-propelled wheeling behavior producing a ballistic rolling by changing its body configuration in 250 ms. The soft robot is composed of three functional components: a soft silicone body molded in a straight and long geometry (bar-like), two independent SMA actuators fixed in parallel respect to the

body allowing its anterior and posterior flexion and a pair of tail skids providing stability. The control system is based on a PW modulation (PWM) driving with high power stimulations in order to achieve a very impulsive response.

Earthworms inspired peristaltic locomotion platforms and contracting robots that exploit antagonistic arrangement and elastic properties of the soft materials they are made of. In [SEO 10], a flexible braided mesh-tube structure is used as a support for SMA springs arranged in an antagonistic manner: circularly to squeeze and elongate the structure and longitudinally to shorten it. Sequential activation of the spring groups leads to peristaltic motion that can be used as a locomotion system. A total of 400 mA current with short pulse time is experimentally found to be the most effective way to activate the springs and the coordination of the segments is obtained by a purposely developed gait controller. Closed-loop control is implemented with the measure of the radius changes by means of Hall effect sensors. In [BYU 06], a different embodiment demonstrates another approach to generate a forward locomotion: a TWE SMA spring located inside a cylindrical silicone bellow. The activation of the spring lets the structure shorten and the bellow delivers the bias force necessary to support the return phase of the alloy. The control system is open-loop and the right timing for cooling and heating has been experimentally set. The robot is completely autonomous and embeds the battery for activation and wireless communication for on/off signals.

Another example of SMAs successfully used as actuation technology for soft robotics is represented by the development of an artificial octopus-like arm. The aim of the work was to develop an octopus-like arm with similar motion capabilities with respect to the natural counterpart by reproducing the fundamental characteristics that make it move so dexterously. The key factor that determines the performance of the octopus arm is the “muscular hydrostat”, a peculiar structure with muscles arranged on longitudinal and transverse planes, which can deform and vary its stiffness with contractions (in [LAS 12], an exhaustive and complete explanation is reported). Applying the principles of the muscular hydrostat led to the design of an artificial muscular hydrostat that is based on a conical braided sheath that is used as the body of the arm and as a support for the SMA-based actuation system (Figure 10.11).

Helicoidal springs have been chosen to meet the requirements of force and deformation. The transverse and longitudinal actuators act as antagonistic elements since the former is able to squeeze the braided sheath causing an elongation of the structure (and consequently of the longitudinal actuators

too) while the latter is able to shorten the arm and to increase the diameter (restoring the transverse actuators). By manufacturing a long structure that counts on several sections such as those shown in Figure 10.11(a), an entire octopus-like arm can be reproduced (Figure 10.11(b)). The combination of the action of the transverse and longitudinal actuators allows the reproduction of the octopus basic movements such as elongation, shortening, bending in any direction (by activating asymmetrically the longitudinal springs) and stiffening (by contracting simultaneously both kinds of actuators). An on/off activation is provided using PWM in an open-loop control. The robot has been designed to work in a marine environment so that an external skin is necessary to separate the external surrounding liquid and the internal controlled environment. Internally, the braided sheath and the actuators are immersed in distilled water that increases the electrical power needed, but guarantees a very fast cooling phase [CIA 12].

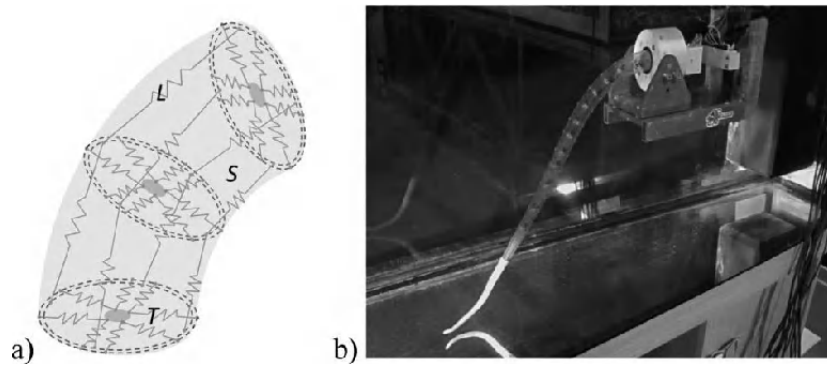


Figure 10.11. a) Schematic design of the artificial muscular hydrostats, with four longitudinal SMA actuators (L), a number of transverse SMA actuators (T) and the support structure (S). b) Entire octopus-like arm

Other than bioinspired and biomimetic robots, the use of SMA in soft robotics is exploited for implementing innovative locomotion strategies. Although rigid platforms such as crawlers and legged robots are often used over rough terrain, an alternative method that exploits the deformability of the robot body is reported in [MAT 08]. The robot is basically composed of a deformable circular shell and eight SMA springs radially arranged inside it. When the springs contract the circle transforms into an ellipse becoming instable so that gravitational potential energy generates a moment around the point of contact between the body and the ground. This causes a body roll

until it achieves a new stable state. The provided activation is on/off through a PWM and with the appropriate voltage pattern a continuous roll is achievable. The same approach has been adopted by Torres-Jara *et al.* [TOR 10] with some modification in the mechanical hardware of the circle and on the disposition of the actuating elements: eight flexible support structures are kept in a circle by SMA sheets. In this case, the deformation of the structure is due to the activation of every laser-cut SMA sheet that exerts a torque between the two flexible structures it connects. Once again the activation method is on/off, but providing direct current and with some safety features to ensure avoiding overheating.

One last example is represented by a very recent and original work where SMA is used in a robotic origami: these kinds of robots are able to fold into three-dimensional (3D) shapes starting from a nominally 2D sheet. In [PAI 11], a stretchable circuitry based on liquid-metal-filled channels in an elastomer substrate is integrated with a tiled origami module actuated by an SMA actuator. In this case, the activation of the alloys is not directly achieved by Joule effect but with the use of external heaters to avoid power losses in the circuitry path.

10.10. Conclusions

SMA's are a relatively new class of smart materials that demonstrated their capability of being successfully used as an alternative to traditional actuation methodologies. Examples of applications in robotics can be found in the biomedical field ranging from active endoscopes and catheters including advanced operational instrumentation to implantable drug delivery devices where a high degree of miniaturization is mandatory, and also in the humanoid robots field where the functional similarity with natural muscles allows the substitution of traditional motors with locally embedded SMA actuators in replicating anthropomorphic limbs. Other commercial applications can be found: in air conditioners to automatically control the air flux temperature without using thermostats and motors; in the regulation of air flow in an electronic oven; in automatic dry boxes, where solenoids used to activate the mechanism that allows the desiccant material to be regenerated can be replaced with an SMA spring; in cooling systems for automobiles, where SMA-driven valves allow the cooling water passing into the radiator only if the engine is warm.

These examples show that at present SMA technology can be used at industry level only in very precise and limited applications (thermal

activation) and this is mainly due to the already mentioned control issues that often do not allow an optimal exploitation of the material properties. But the still growing number of publications in the field (both from a material science and engineering point of view) demonstrates a high interest in the possibility of substituting traditional actuation technologies with SMAs, especially in the applications where space and weight are the main constraints. In this sense, soft robotics is a paradigmatic example: in a research area that is based on the necessity of achieving adaptive, flexible interaction also with unexpected environments and being intrinsically safe, SMAs are particularly appreciated as demonstrated by the reported application examples. Alternative solutions are represented by cable-driven systems and fluidic actuators, but their usability is limited by the necessity of bulky external components necessary to drive the systems such as electromagnetic motors and fluidic pumps, respectively. Other more innovative solutions are under evaluation and they seem to be very promising: electro-active polymers and shape memory polymers are the most representative examples, but the maturity of such technologies does not allow a wide applicability yet. Thus, at present, SMAs represent one of the most effective soft/flexible actuator technologies available. However, soft robotics is a very new research area; thus, its potential has not been totally exploited yet. But ongoing progress anticipates a future trend where robotic systems will be complemented by soft technologies. Inherent mechanical properties of the soft structure will allow more resilient performances and adaptive behaviors reducing electrical and algorithmic complexity. From this point of view, the combination of soft/flexible materials and SMAs could give a very important contribution in the development of robotic systems following a mechatronics approach, as these soft robotics technologies will be crucial component in future robotic applications.

10.11. Bibliography

- [ALB 08] ALBU-SCHAFER A., EIBERGER O., GREBENSTEIN M., *et al.*, “Soft robotics”, *IEEE Robotics and Automation Magazine*, vol. 15, pp. 20–30, 2008.
- [BYU 06] BYUNGKYU K., MOON G.L., YOUNG P.L., *et al.*, “An earthworm-like micro robot using shape memory alloy actuator”, *Sensors and Actuators A*, vol. 125, pp. 429–437, 2006.
- [CIA 12] CIANCHETTI M., FOLLADOR M., MAZZOLAI B., *et al.*, “Design and development of a soft robotic octopus arm exploiting embodied intelligence”, 14–18 May, *International Conference on Robotics and Automation (ICRA) 2012*, St. Paul, MN, pp. 5271–5276, 2012.

- [DUE 90] DUERING T.W., MELTON K.N., STÖCKEL D., *et al.*, *Engineering Aspects of Shape Memory Alloys*, Butterworth-Heinemann, London, 1990.
- [HAB 07] HABIB M.K., “Mechatronics: a unifying interdisciplinary and intelligent engineering paradigm”, *IEEE Industrial Electronics Magazine*, vol. 1, no. 2, pp. 12–24, Summer 2007.
- [HAS 98] HASEGAWA T., MAJIMA S., “A control system to compensate the hysteresis by Preisach model of SMA actuator”, *Proceedings of IEEE International Symposium on Micromechatronics and Human Science*, Nagoya, Japan, pp. 171–176, 1998.
- [LAS 12] LASCHI C., MAZZOLAI B., CIANCHETTI M., *et al.*, “A soft robot arm inspired by the octopus”, *Advanced Robotics* (Special Issue on Soft Robotics), vol. 26, no. 7, pp. 709–727, 2012.
- [LEE 08] LEESTER-SCHÄDEL M., HOXHOLD B., LESCHE S., *et al.*, “Micro actuators on the basis of thin SMA foils”, *Microsystem Technologies*, vol. 14, no. 4, pp. 697–674, 2008.
- [LIN 11] LIN H.T., LEISK G.G., TRIMMER B., “GoQBot: a caterpillar-inspired soft-bodied rolling robot”, *Bioinspiration and Biomimetics*, vol. 6, 026007, 2011.
- [MA 03] MA N., SONG G., “Control of shape memory alloy actuator using pulse width modulation”, *Smart Materials and Structures*, vol. 12, pp. 712–719, 2003.
- [MAT 08] MATSUMOTO Y., NAKANISHI H., HIRAI S., “Rolling locomotion of a deformable soft robot with built-in power source”, *International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines (CLAWAR)*, 8–10 September, Coimbra, Portugal, pp. 365–372, 2008.
- [PAI 11] PAIK J.K., KRAMER R.K., WOOD R.J., “Stretchable circuits and sensors for robotic origami”, *International Conference on Intelligent Robots and Systems (IROS)*, 25–30 September, San Francisco, CA, pp. 414–420, 2011.
- [PAI 06] PAIVA A., SAVI M.A., “An overview of constitutive models for shape memory alloys”, *Mathematical Problems in Engineering*, vol. 2006, pp. 1–30, 2006.
- [SEO 10] SEOK S., ONAL C.D., WOOD R., *et al.*, “Peristaltic locomotion with antagonistic actuators in soft robotics”, 3–8 May, *International Conference on Robotics and Automation (ICRA) 2010 Anchorage, AK*, pp. 1228–1233, 2010.
- [SON 03] SONG G., CHAUDHRY V., BATUR C., “Precision tracking control of shape memory alloy actuators using neural networks and sliding-mode based robust controller”, *Smart Materials and Structures*, vol. 12, pp. 223–231, 2003.
- [TOR 10] TORRES-JARA E., GILPIN K., KARGES J., *et al.*, “Compliant modular shape memory alloy actuators”, *IEEE Robotics and Automation Magazine*, vol. 17, no. 4, pp. 78–87, 2010.

- [TRI 08] TRIVEDI D., RAHN C., KIER W., *et al.*, “Soft robotics: biological inspiration, state of the art, and future research”, *Applied Bionics and Biomechanics*, vol. 5, pp. 99–117, 2008.
- [WEL 08] WELTY J.R., WICKS C.E., WILSON R.E., *et al.*, *Fundamentals of Momentum, Heat and Mass Transfer*, 5th ed., John Wiley & Sons Ltd, 2008.
- [WOL 95] WOLF R.H., HEUER A.H., “TiNi (Shape Memory) films on silicon for MEMS applications”, *Journal of Microelectromechanical Systems*, vol. 4, no. 4, pp. 206–212, 1995.
- [WU 01] WU M.H., “Fabrication of nitinol materials and components”, *International Conference on Shape Memory and Superelastic Technologies (SMST)*, Kunming, China, pp. 285–292, 2001.

Chapter 11

Tuned Modified Transpose Jacobian Control of Robotic Systems

Control of robotic systems is a challenging task due to extreme nonlinearities and inherent coupling in system dynamics. Model-based (MB) control approaches can guarantee asymptotic convergence of the tracking error. However, implementing an MB algorithm, in addition to *a priori* knowledge of the system properties, requires computational power that may not be available. On the other hand, non-MB control approaches can provide simple alternatives for real-time implementations. The modified transpose Jacobian (MTJ) algorithm has been proposed based on an approximated feedback linearization approach, in which there is no need for *a priori* knowledge of the plant dynamics. This chapter presents a tuned MTJ (TMTJ) control algorithm that improves and facilitates the implementation of the MTJ algorithm by proper tuning of the switching gain matrix. This proposed control algorithm eliminates heuristic tuning for sensitivity thresholds of switching factor in the MTJ algorithm that will be useful for online control implementations. Obtained results show the merits of this new TMTJ controller as its performance is improved compared to that of the original MTJ algorithm even in the presence of significant disturbances and noises. Therefore, extremely low computational needs, without requiring *a priori* knowledge of the system dynamics, makes the proposed TMTJ

algorithm a suitable candidate for position control of robotic systems in real-time practical implementations.

11.1. Introduction

Nonlinearity and complicated coupling in the system dynamics of mechanical manipulators have created challenging control problems. Hence, various algorithms have been proposed to tackle these problems, including adaptive control algorithms [SLO 87, TAI 00], time-delay control [YOU 87], motion-rate control [UME 89, KEL 05], artificial neural networks and fuzzy control [DOM 04, MBE 05, MEG 05, STE 04]. In practice, model-based (MB) algorithms can hardly yield a satisfactory solution due to their need to exact mathematical models of the system, which may be tremendously difficult to obtain in many real cases. On the other hand, among all these algorithms, those with fewer computational operations are more preferable for real-time implementations.

Transpose Jacobian (TJ) control is one of the simplest algorithms used to control motion of robotic manipulators, which has been arrived at intuitively [CRA 89]. In the case of using an approximate Jacobian, it has been shown that the damping matrix and the position gain matrix of this controller play an important role in system stability [MIY 88]. Apparently, the algorithm can be applied to redundant manipulators as shown by [ASA 93], and as discussed by [CHI 91] it does not fail when a singularity occurs. Hootsmans and Dubowsky [HOO 91] have developed an extended Jacobian transpose control algorithm to improve the performance of mobile manipulator systems. Subsequently, to fulfill simplicity requirements, Bevely *et al.* [BEV 00] have developed simplified Cartesian computed torque (SCCT) control algorithms for highly geared climbing robots.

The performance of TJ-based algorithms has been experimentally compared to those of different algorithms using unit quaternions on a direct-drive spherical wrist given by [GAR 02]. Papadopoulos and Moosavian [PAP 95] have compared the performance of this simple algorithm to those of various MB algorithms. Both experimental and simulation results show the merits of the TJ algorithm in controlling of highly nonlinear and complex systems with multiple degrees of freedom (DoFs), motivating further work on this algorithm. However, since the TJ is not dynamics-based, poor performance may result in fast trajectory tracking. Use of high gains can deteriorate performance seriously in the presence of feedback measurement

noise. Another drawback is that there is no formal method of selecting its control gains, and a heuristic selection of gains makes it difficult to apply.

Hence, the modified transpose Jacobian (MTJ) has been proposed, which yields an improved performance over the standard algorithm, by employing stored data of the previous time step control command [MOO 97, MOO 07]. The MTJ algorithm is based on an approximation of feedback linearization methods, with no need for *a priori* knowledge of the plant dynamics terms. Its performance is comparable to that of MB algorithms, but requires reduced computational burden. Unlike the standard TJ, this algorithm works well in high-speed tracking tasks. In addition, controller gains can be selected in a systematic rather than a heuristic way, while higher gains and sensitivity to noise are avoided. The MTJ algorithm has been exploited for control of various systems; for instance, it has been recently implemented on a tractor-trailer wheeled mobile robot [KEY]. This is a modular robotic system that consists of a tractor module towing a passive trailer. Experimental results show the effectiveness of the proposed controller based on the MTJ algorithm. Nevertheless, the sensitivity thresholds of switching factor have to be assigned specifically for each robotic system before the tracking task. This can be a deficiency of the MTJ control law and needs to be resolved, which is the main focus of this study.

This chapter presents the tuned MTJ (TMTJ) control law that yields an improved applicability over the previously modified algorithm by tuning the switching gain matrix. Performance of the TMTJ algorithm will be compared to that of MTJ algorithm. Obtained results reveal the efficiency of the proposed algorithm even in the presence of influential disturbances, noises and substantial increase of trajectory frequency without requiring delicate selection of sensitivity thresholds.

11.2. TMTJ control law

11.2.1. Feedback linearization approach

Using the expressions for the kinetic and potential energy, and applying Lagrange's equations for a robotic system, the dynamics model can be obtained as follows:

$$H(q)\ddot{q} + C(q, \dot{q}) = Q \quad [11.1]$$

where $q \in \mathfrak{R}^n$ is the vector of generalized coordinates, $H \in \mathfrak{R}^{n \times n}$ is the inertia matrix and $C \in \mathfrak{R}^n$ is the centripetal, coriolis and gravitational forces.

The output velocities $\dot{\hat{q}}$ are obtained from the generalized velocities $\dot{q}$ using a Jacobian matrix, $J(q)$, as:

$$\dot{\hat{q}} = J(q) \dot{q} \quad [11.2]$$

Assuming that $J(q)$ is square and non-singular, equation [11.1] can be written in terms of the output variables as follows:

$$\hat{H}(q) \ddot{\hat{q}} + \hat{C}(q, \dot{\hat{q}}) = \hat{Q} \quad [11.3a]$$

where

$$\hat{H} = J^{-T} H J^{-1}, \quad \hat{C} = J^{-T} C - \hat{H} \dot{J} \dot{q}, \quad \hat{Q} = J^{-T} Q \quad [11.3b]$$

To control such a system, a model-based control law (MB algorithm), such as

$$Q = J^T (\hat{H}u + \hat{C}) \quad [11.4]$$

can be applied, where u is an auxiliary control signal. A usual assumption associated with this control law is that the system geometric and mass properties are known. This control law linearizes and decouples the system equations to a set of second-order differential equations:

$$\ddot{\hat{q}} = u \quad [11.5]$$

If u is computed such that

$$u = k_p e + k_d \dot{e} + \ddot{q}_{des} \quad [11.6]$$

where k_p and k_d are positive definite gain matrices and e is the tracking error defined as:

$$e = \hat{q}_{des} - \hat{q} \quad [11.7]$$

then the control law given by equation [11.4] guarantees asymptotic convergence of the tracking error e . Implementing an MB algorithm, in addition to *a priori* knowledge of the system properties, requires computational power that may not be available.

11.2.2. Transpose Jacobian algorithm

The TJ control law is a computationally simple algorithm, which has been arrived at intuitively. The task error vector and its rate, both multiplied by relatively high gains and by the Jacobian transpose matrix, result in commands that push the end-effector (EE) in a direction that tends to reduce the tracking error. The TJ controller can be presented as follows [CRA 89]:

$$\dot{Q} = J^T(q) \{k_p e + k_d \dot{e}\} \quad [11.8]$$

However, since TJ control law is not dynamics-based, poor performance may occur in applications where high-speed tracking is required. Use of high gains makes this problem more serious, especially in the presence of noise.

11.2.3. Modified transpose Jacobian algorithm

To achieve both precision and simplicity, the TJ control law has been modified by inclusion of a term representing the system dynamics at the time steps when the system experiences low errors. As shown in Figure 11.1, the MTJ approximates a feedback linearization solution by using stored data of the control command in the previous time step.

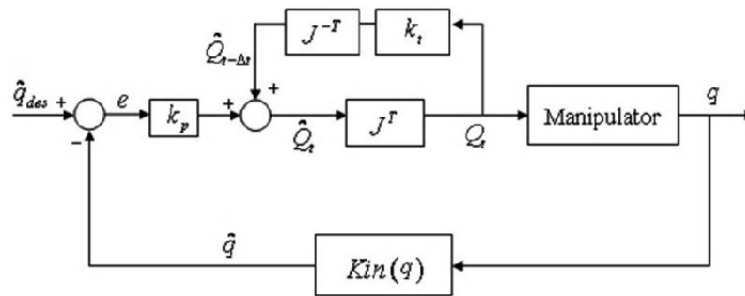


Figure 11.1. The block diagram of the MTJ control law

The MTJ control law is obtained as [MOO 07]:

$$Q = J^T(q) \{k_p e + k_d \dot{e} + h(t)\} \quad [11.9]$$

where $h(t)$ is the product of the control command in the previous time step and a switching gain matrix k_t that can be presented as follows:

$$h(t) = k_t \hat{Q}_{t-\Delta} \quad [11.10]$$

$$k_{t,ii} = \exp \left(- \left(\frac{|e_i|}{e_{\max,i}} + \frac{|\dot{e}_i|}{\dot{e}_{\max,i}} \right) \right) \quad [11.11]$$

where $e_{\max}$ and $\dot{e}_{\max}$ represent sensitivity thresholds of the MTJ controller. Therefore, with proper selection of these sensitivity thresholds, so that the modifying term is properly activated (i.e. $k_t \approx 1.0$), and small time steps, the following error equation results [MOO 07]:

$$k_{d,ii} \dot{e}_i + k_{p,ii} e_i \equiv 0 \quad [11.12]$$

which means the error dynamics will be governed by appropriate controller gains selection. It should be mentioned that the expression similar to equation [11.11] has been successfully employed to regulate sliding mode controllers in [MOO 04] for chattering elimination. Note that for better tracking, higher gains are required for the TJ algorithm and these lead to poor noise rejection characteristics. Also, high-frequency inputs can excite flexible system modes, and consequently decrease the accuracy, and the useful life of a system. Hence, it is confirmed that using high gains is not a viable option. On the other hand, the MTJ algorithm, by being an approximation of a feedback linearization algorithm, does not require high gains, or a high computational power, while its performance is comparable to that of the MB algorithms [MOO 07].

Considering equations [11.9]–[11.11], it can be deduced that for an N DoF system, calculation of the MTJ law requires $3N^2 + N + 2$ multiplications and $3N^2 - N + 1$ additions. Compared to the results presented in Table 11.1, these are almost the same as those for the TJ law and still significantly less compared to those needed for implementing the MB laws. Note that it is assumed that the inverse of the Jacobian matrix and its time derivative, which are required for implementing MB algorithms, are available symbolically, and hence these computations are not counted in Table 11.1. However, this

comparison in terms of required computational effort reveals the efficiency of the TJ and the MTJ algorithms.

Algorithm	Multiplication	Additions
TJ	$3N^2$	$3N^2 - 2N$
MTJ	$3N^2 + 2$	$3N^2 - N + 1$
MB	$2N^3 + 7N^2$	$2N + 5N^2 - 4N$

Table 11.1. Comparison of the required computational operations

The above analysis reveals the simplicity (concerning *a priori* the knowledge requirement of system dynamics) and efficiency (in terms of the required computational effort) of both the standard TJ and the new MTJ law compared to the MB algorithms. In addition, the MTJ yields approximately linearized error dynamics and therefore an improved performance over the standard TJ algorithm. Stability analysis of the developed MTJ algorithm, based on Lyapunov's theorems, shows that both the standard and the MTJ algorithms are globally asymptotically stable [MOO 97].

It should be noted that the MTJ law is a position control algorithm that yields an improved performance over the standard algorithm. However, to manipulate an object, application of force/impedance control laws will be required that are usually MB algorithms. For instance, the multiple impedance control (MIC) is an MB algorithm that requires knowledge of the system dynamics [MOO 05, MOO 10, RAS 10]. On the other hand, even if the system dynamics is perfectly known, its computation may require considerable process time at each step for implementing the control law. Based on the MTJ control approach proposed earlier, the MIC law has been recently modified to be implemented without using system dynamics as non-MB MIC (NMIC) [MOO 08]. Therefore, this NMIC law is a more realistic algorithm for online computations in cooperating robotic systems.

11.2.4. Tuned modified transpose Jacobian algorithm

The sensitivity thresholds of a switching gain matrix may have different values in each robotic system. Therefore, they must be determined by trial-and-error, prior to a tracking task being performed. This can be a deficiency of the MTJ control law that is improved here as a TMTJ algorithm. To

overcome this weakness, the sensitivity thresholds of a switching gain matrix can be computed according to the proposition 11.1.

PROPOSITION 11.1.– *If k_t is computed as bellow, the feedback linearization can be smoothly approximated as:*

$$k_{tii} = \exp \left(- \left| \frac{k_{p\ ii} e_i + k_{d\ ii} \dot{e}_i}{r \hat{Q}_{t-\Delta t i}} \right|^s \right) \quad [11.13]$$

PROOF.– Considering the above switching gain matrix elements, $0 \leq k_{tii} \leq 1$, where $k_{p\ ii} e_i + k_{d\ ii} \dot{e}_i$ can be determined from the MTJ law, equations [11.9] and [11.10], and substituting the result into [11.13], we obtain:

$$k_{tii} = \exp \left(- \left| \frac{\hat{Q}_{ti} - k_{tii} \hat{Q}_{t-\Delta t i}}{r \hat{Q}_{t-\Delta t i}} \right|^s \right) \quad [11.14]$$

For simplicity, defining $\mu \geq 0$ as follows:

$$\mu = \left| \frac{\hat{Q}_{ti} - k_{tii} \hat{Q}_{t-\Delta t i}}{\hat{Q}_{t-\Delta t i}} \right| \xrightarrow{(b)} k_{tii} = \exp \left(- \frac{\mu^s}{r^s} \right) \quad [11.15]$$

Then, two cases will exist as discussed below.

$$\text{Case 1: } \frac{\hat{Q}_{ti} - k_{tii} \hat{Q}_{t-\Delta t i}}{\hat{Q}_{t-\Delta t i}} \geq 0$$

Relative difference of $\hat{Q}_{ti}$ is defined from the definition of μ , equation [11.15] and the aforementioned inequality.

$$E_{\hat{Q}_{ti}}^{rel} = \frac{\hat{Q}_{ti} - \hat{Q}_{t-\Delta t i}}{\hat{Q}_{ti}} = 1 - \frac{1}{\mu + k_{tii}} \quad [11.16]$$

Substituting equation [11.15] into [11.16], $E_{\hat{Q}_{ti}}^{rel}$ can be rewritten as:

$$E_{\hat{Q}_{ti}}^{rel} = 1 - \frac{1}{r(-\ln(k_{tii}))^{1/s} + k_{tii}} \quad [11.17]$$

Claim I: If $s > 1$ and $r > \lambda_{(s)}s$, then:

$$\forall k_{tii} \in (0,1) \Rightarrow 0 < E_{\dot{q}_i}^{rel} \text{ (as strictly decreasing)} < 1$$

$$\text{if } k_{tii} \rightarrow 1^- \Rightarrow \frac{\partial k_{tii}}{\partial E_{\dot{q}_i}^{rel}} \rightarrow 0$$

PROOF.— Suppose $r, s > 0$, thus equation [11.17] yields:

$$\text{when } k_{tii} \rightarrow 0^+ \Rightarrow E_{\dot{q}_i}^{rel} \rightarrow 1^-$$

$$\text{when } k_{tii} \rightarrow 1^- \Rightarrow E_{\dot{q}_i}^{rel} \rightarrow 0^+$$

Therefore, it is sufficient to find a condition on which $\partial E_{\dot{q}_i}^{rel} / \partial k_{tii} < 0$ is satisfied. Using derivation of $E_{\dot{q}_i}^{rel}$ in term of k_{tii} :

$$\frac{\partial E_{\dot{q}_i}^{rel}}{\partial k_{tii}} = \frac{-\frac{r}{s} \times \frac{1/k_{tii}}{(-\ln(k_{tii}))^{1-1/s}} + 1}{\left(r(-\ln(k_{tii}))^{1/s} + k_{tii}\right)^2} \quad [11.18]$$

It is seen that the gradient of $E_{\dot{q}_i}^{rel}$ in term of k_{tii} is negative if:

$$\frac{r}{s} \times \frac{1/k_{tii}}{(-\ln(k_{tii}))^{1-1/s}} > 1$$

The minimum value of an LHS of the above inequality is achieved when $k_{tii}(-\ln(k_{tii}))^{1-1/s}$ meets its maximum value as happened in the point $k_{tii} = \exp(1/s - 1)$. Substituting this point into the inequality results in $r > \exp(1/s - 1)(1 - 1/s)^{(1-1/s)}s$.

According to $r, s > 0$, when k_{tii} tends to 1^- , we can write:

$$\lim_{k_{tii} \rightarrow 1^-} \frac{\partial E_{\dot{q}_i}^{rel}}{\partial k_{tii}} = \frac{-\frac{r}{s} \times \frac{1/1}{(\ln(1^+))^{1-1/s}} + 1}{\left(r(\ln(1^+))^{1/s} + 1\right)^2} = -\frac{r}{s} \times \frac{1/1}{(0^+)^{1-1/s}} + 1 \quad [11.19]$$

If $1-1/s > 0$ or $s > 1$, thus $\partial E_{\hat{Q}_i}^{rel} / \partial k_{tii} \rightarrow -\infty$ that leads to $\partial k_{tii} / \partial E_{\hat{Q}_i}^{rel} \rightarrow 0$, then claim I is proven.

So, using different r and s values to plot a $k_{tii}(E_{\hat{Q}_i}^{rel})$ diagram, the suitable non-dimensional parameters can be determined. Some simulations reveal that choosing the conditions from a range approximately 0.5 for $E_{\hat{Q}_i}^{rel}$ at $k_{tii} = 0.5$ can be proper selections and their results are nearly the same.

$$\text{Case 2: } \frac{\hat{Q}_{ti} - k_{tii}\hat{Q}_{t-\Delta t i}}{\hat{Q}_{t-\Delta t i}} < 0$$

Relative difference of $\hat{Q}_{t-\Delta t i}$ is defined from the definition of μ , equation [11.15] and the above-mentioned inequality.

$$E_{\hat{Q}_{t-\Delta t i}}^{rel} = \frac{\hat{Q}_{t-\Delta t i} - \hat{Q}_{ti}}{\hat{Q}_{t-\Delta t i}} = 1 + \mu - k_{tii} \quad [11.20]$$

Substituting equation [11.15] into [11.20], $E_{\hat{Q}_{t-\Delta t i}}^{rel}$ can be rewritten as:

$$E_{\hat{Q}_{t-\Delta t i}}^{rel} = 1 + r(-\ln(k_{tii}))^{1/s} - k_{tii} \quad [11.21]$$

Claim II: If $s, r > 0$, then:

$$\forall k_{tii} \in (0, 1) \Rightarrow E_{\hat{Q}_{t-\Delta t i}}^{rel} > 0 \text{ is strictly decreasing.}$$

PROOF.— Suppose $r, s > 0$, thus equation [11.20] yields:

$$\text{when } k_{tii} \rightarrow 0^+ \Rightarrow E_{\hat{Q}_{t-\Delta t i}}^{rel} \rightarrow +\infty$$

$$\text{when } k_{tii} \rightarrow 1^- \Rightarrow E_{\hat{Q}_{t-\Delta t i}}^{rel} \rightarrow 0^+$$

Therefore, it is sufficient to find a condition on which $\partial E_{\hat{Q}_{t-\Delta t i}}^{rel} / \partial k_{tii} < 0$ is satisfied. Using derivation of $E_{\hat{Q}_{t-\Delta t i}}^{rel}$ in terms of k_{tii} :

$$\frac{\partial E_{\hat{Q}_{t-\Delta t i}}^{rel}}{\partial k_{tii}} = -\frac{r}{sk}(-\ln(k_{tii}))^{(1/s)-1} - 1 \quad [11.22]$$

It can be easily found that by the condition of $r, s > 0$, $\partial E_{\hat{Q}_{t-\Delta t}}^{rel} / \partial k_{tii}$ is negative and claim II is proven.

According to claim II, it can be understood that the values employed in case 1 for r and s are appropriate for case 2 too.

It should be mentioned that it is possible that the difference between two consecutive switching factors can be naturally large due to the changing control law from TJ to MTJ algorithm and approaching $\hat{Q}_{t-\Delta t}$ to zero. Hence, the following switching gain matrix elements are filtered by a Butterworth low-pass filter as:

$$k_{tii}^{filtered} = \text{Butterworth (low-pass)} \times k_{tii} \quad [11.23]$$

where cutoff frequency is chosen relevant to the step size.

Note that factor k_t is initially taken equal to zero, resulting in a TJ control law at the first time step. Stability analysis of the MTJ algorithm, based on Lyapunov's theorems, shows that both the standard and the MTJ algorithms are globally asymptotically stable [MOO 97]. This analysis obviously justifies the same result for the proposed TMTJ algorithm.

11.3. Obtained results and discussions

In this section, the performance of the TMTJ control algorithm as given by equations [11.9], [11.10] and [11.13] is evaluated and compared to that of the MTJ algorithm, as given by equations [11.9], [11.10] and [11.11]. Two different systems as fixed and mobile base manipulators are considered for this comparison.

11.3.1. Fixed base manipulator

To focus on algorithmic aspects, a simple two-link planar manipulator as shown in Figure 11.2(a) is considered under various conditions. Performing high-speed tracking task, disturbance and noise rejection characteristics of the proposed TMTJ algorithm is investigated in these simulations and compared to that of alternative algorithm.

The task is tracking a trajectory, defined by:

$$\begin{aligned} x_{des} &= \sqrt{l_1^2 + l_2^2} \cos(\omega t + \pi/4) + 0.1 \sin(5\omega t) \\ y_{des} &= \sqrt{l_1^2 + l_2^2} \sin(\omega t + \pi/4) + 0.1 \sin(5\omega t) \end{aligned} \quad [11.24]$$

This trajectory corresponds to a perturbed circular path (see Figure 11.2(b)). The motion speed along the path can be selected by setting the cyclical frequency ω .

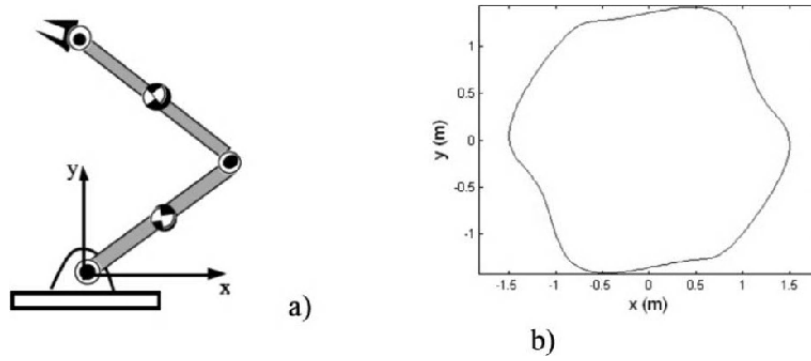


Figure 11.2. a) The manipulator and b) the desired tracking path

For the manipulator shown in Figure 11.2(a), the mass properties of the system are $m_1 = 4.0$ kg, $I_1 = 0.333$ kg/m², $m_2 = 3.0$ kg and $I_2 = 0.30$ kg/m², and the link lengths are $l_1 = l_2 = 1$ m. The initial conditions for joint angles and derivatives are:

$$(q_1(0), q_2(0), \dot{q}_1(0), \dot{q}_2(0)) = (0.03, \pi/2, 1.5, -1.0) \text{ (rad, rad/s)}$$

which correspond to some initial position and velocity errors. The sensitivity thresholds for the MTJ algorithm, $e_{\max}$ and $\dot{e}_{\max}$, in equation [11.11], are equal to 1 m and 10 m/s, respectively. Also, r and s parameters are set to 1.5 for both in TMTJ. The time step Δt_i is held constant and equal to 10.0 ms in this study. To establish a fair comparison, the gains for the algorithms under comparison should be selected such that the peaks of the required joint

torques become approximately equal. Herein, choosing similar gains for both algorithms satisfies this condition. The fourth-order Runge–Kutta method for solving differential equations is used in all simulations.

The performance of the MTJ and TMTJ algorithms, in terms of the end-point error in relatively high-speed tracking task ($\omega = 1$ rad/s), is compared in Figure 11.3. For both MTJ and TMTJ algorithms, $k_p = \text{diag}(30,30)$ and $k_d = \text{diag}(60,60)$ are properly selected based on error dynamics of equation [11.12], which are used for all simulations in this case.

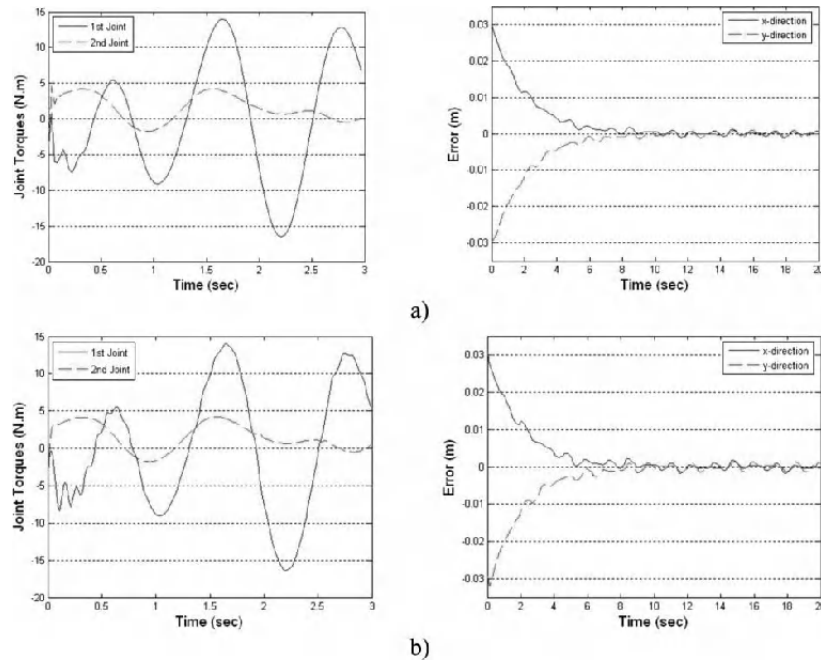


Figure 11.3. (Left) Joint torques and (right) end-effector errors:
a) MTJ and b) TMTJ

It can be seen that the input torques in the TMTJ law have somewhat more fluctuation than that of MTJ control law. The end-point error is almost the same for both algorithms, while TMTJ does not need heuristic selection of the sensitivity thresholds, $e_{\max}$ and $\dot{e}_{\max}$ in equation [11.11]. Changing of r

and s parameters in the range, pointed out in the proof of proposition 1, has made no considerable differences in the results. It should be mentioned that the total energy consumption of each algorithm for performing this task, given by the time integral of $\sum_{i=1}^2 |Q_{a,i} \dot{q}_i|$, is approximately the same.

11.3.1.1. Disturbance rejection characteristics

To show the capability of the proposed TMTJ control law for disturbance effect elimination, it is assumed that the input torques are perturbed by 20% with respect to the true values. Figure 11.4 shows that TMTJ and MTJ are similarly capable to preserve the trend of tracking errors elimination by more fluctuating in input torques in the presence of influential disturbances.

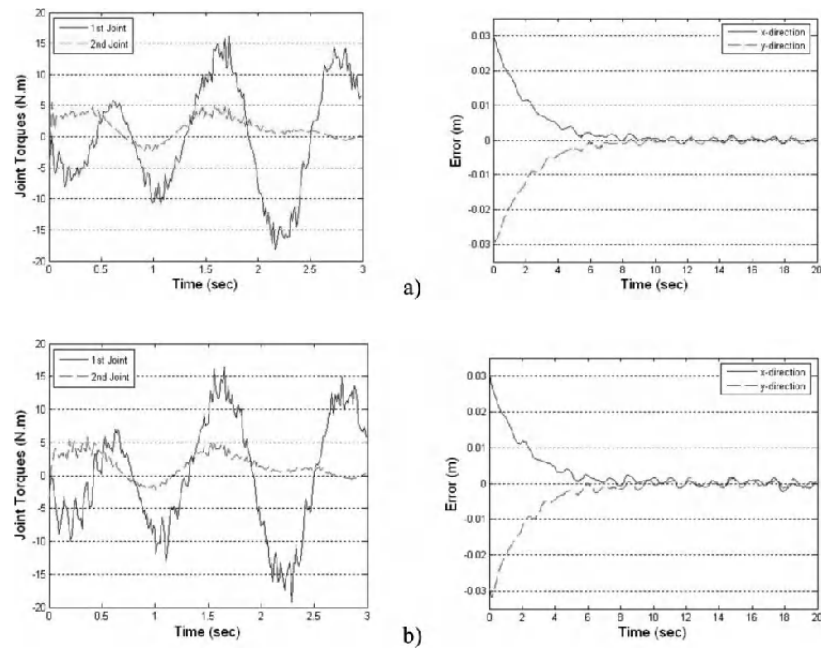


Figure 11.4. (Left) Joint torques and (right) EE errors in the presence of disturbances: a) MTJ and b) TMTJ

11.3.1.2. Noise rejection characteristics

In practice, noise effects deteriorate available feedback. Therefore, we should examine the noise rejection capabilities of would be implemented algorithms. Here, it is supposed that measurements of joint angles rates are corrupted by white noise whose amplitude is 1% of the signal magnitude. As shown in Figure 11.5, the tracking errors of TMTJ are comparable with those of an MTJ algorithm while the required torques for both algorithms are of reasonably equal magnitude.

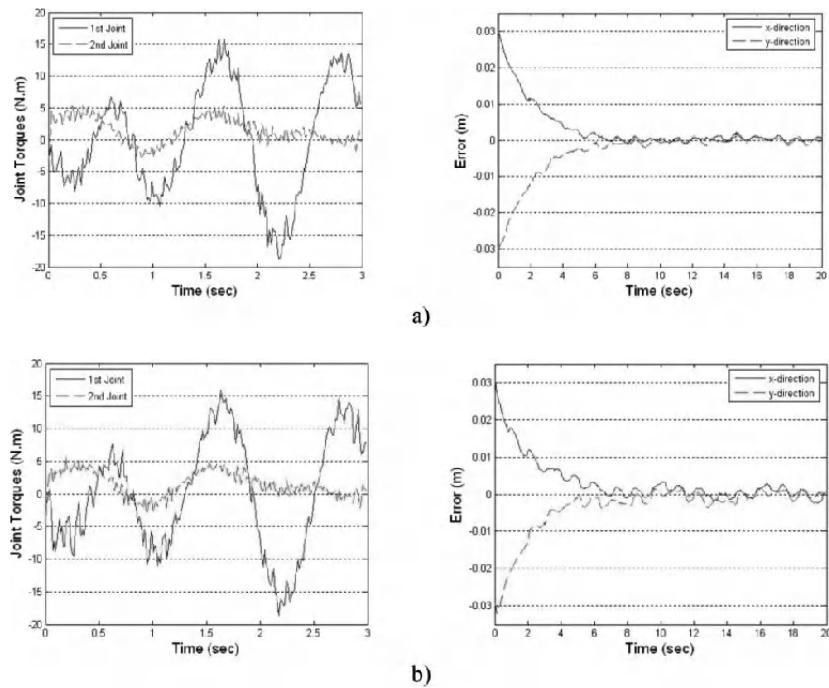


Figure 11.5. (Left) Joint torques and (right) EE errors in the presence of noises: a) MTJ and b) TMTJ

11.3.2. Mobile base manipulator

To show the advantages of the proposed TMTJ compared to the MTJ algorithm, a mobile two-link planar manipulator is considered (Figure 11.6).

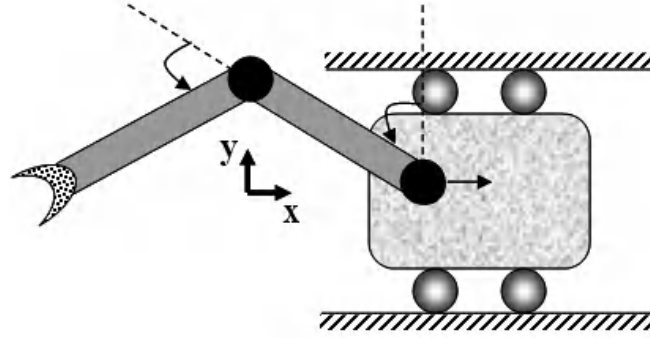


Figure 11.6. *The mobile base manipulator*

The task is defined for the end-effector as tracking a trajectory given by:

$$\begin{aligned} x_{des} &= \cos(\omega t) \\ y_{des} &= 0.1 \sin(\omega t) \\ \theta_{des} &= 0.12 \cos(\omega t) + \frac{5\pi}{6} \end{aligned} \quad [11.25]$$

This trajectory corresponds to an elliptic path. For the manipulator shown in Figure 11.6, the mass of the base is $m_1 = 5.0$ kg and mechanical properties of the arms are the same as for the previous fixed base manipulator. The initial conditions for joint angles and derivatives are:

$$(q_1(0), q_2(0), q_3(0), \dot{q}_1(0), \dot{q}_2(0), \dot{q}_3(0)) = (0.03, \pi/2, 1.5, -1.0) \text{ (rad, rad/s)}$$

which correspond to some initial position and velocity errors.

The sensitivity thresholds for the MTJ algorithm, $e_{\max}$ and $\dot{e}_{\max}$ in equation [11.11], are equal to 1 m and 10 m/s for x and y directions, respectively, and 3 rad and 30 rad/s for θ . Also, r and s parameters are set equal to 1.5 as was priorly used for previous case. Choosing similar gains $k_p = \text{diag}(30, 30)$ and $k_d = \text{diag}(60, 60)$ for both algorithms satisfies a fair comparison condition, that is total energy consumption and the peaks of the required joint torques are approximately the same for both algorithms. First, the end-effector error is compared in Figure 11.7 for a tracking task with constant frequency of $\omega = 1$ rad/s. It can be seen that tracking errors are similar for both algorithms, if not better for the TMTJ.

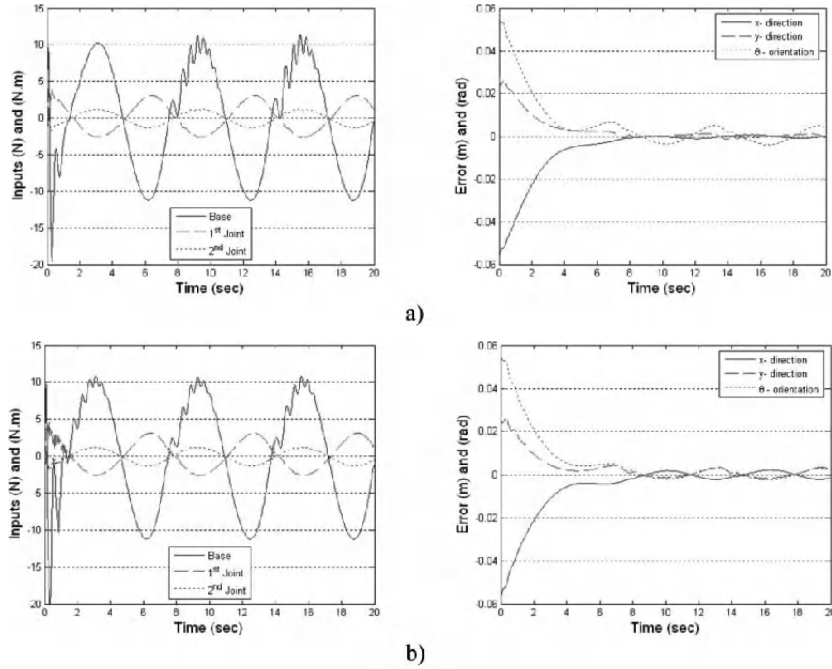


Figure 11.7. (Left) Joint torques and (right) end-effector errors:
a) MTJ and b) TMTJ

11.3.2.1. Increasing tracking frequency

To show the merits of the proposed TMTJ algorithm, tracking frequency ω is increased from 1 to 3 rad/s, using a simple exponential function:

$$\omega = 3 - 2e^{-t} \quad [11.26]$$

As it is shown in Figure 11.8, tracking errors for the MTJ are almost twice those of the TMTJ. The reason is that the sensitivity thresholds for the MTJ cannot be well tuned when the frequency ω is being developed. Also, it should be noted that r and s parameters in the TMTJ can be adapted with change in ω . As seen in the left side of Figure 11.8, the TMTJ algorithm demands slightly different input force and torques to preserve a better tracking performance.

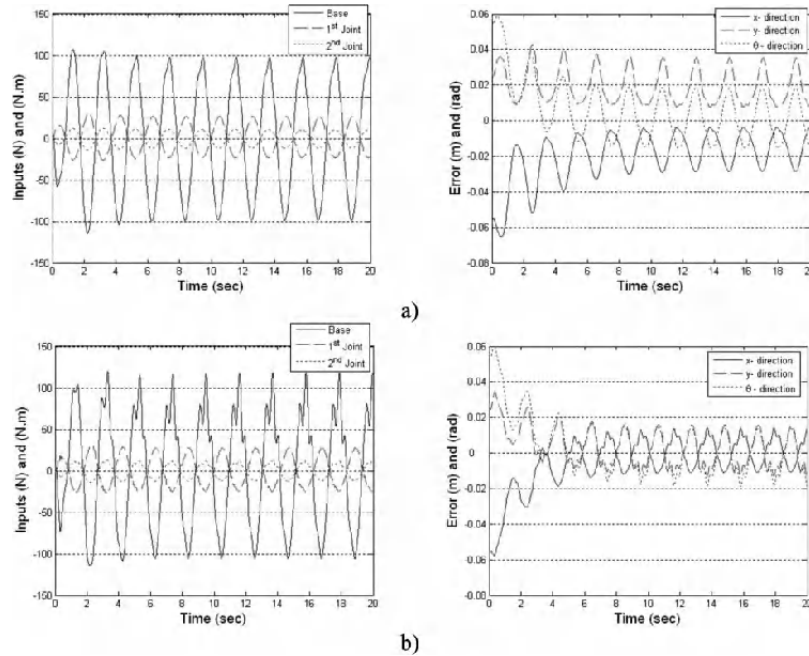


Figure 11.8. (Left) Joint torques and (right) EE errors with the increased frequency: a) MTJ and b) TMTJ

11.4. Conclusions

This chapter presented a new TMTJ control algorithm that improves the applicability of the MTJ algorithm, by eliminating the requirement of the delicate selection of the switching gain matrix for online control purposes. Obtained results reveal substantial merits of the proposed TMTJ controller as its performance is improved compared to the original MTJ algorithm even in the presence of significant disturbances and noises, and also in the increase of trajectory frequency. It was already shown that the performance of the MTJ controller is comparable to that of a perfect MB algorithm, with the advantage that less computational power is needed. Extremely low computational needs, without requiring *a priori* knowledge of the system dynamics, makes the proposed TMTJ algorithm a promising alternative for position control of robotic systems in practice. Therefore, the proposed TMTJ algorithm can be considered as a good candidate for the control of industrial robots, where simple efficient algorithms are vastly preferred to complicated theoretical algorithms that usually require huge computations.

11.5. Bibliography

- [ASA 93] ASARI Y., SATO H., YOSHIMI T., *et al.*, “Development of a model-based remote maintenance robot system (IV) – a practical stiffness control method for redundant robot arm”, *Proceedings of the First International Conference on Intelligent Robots and Systems*, Yokohama, Japan, 26–30 July 1993.
- [BEV 00] BEVLY D., DUBOWSKY S., MAVROIDIS C., “A simplified Cartesian-computed torque controller for highly geared systems and its application to an experimental climbing robot”, *Journal of Dynamic Systems, Measurement, and Control*, vol. 122, pp. 27–32, 2000.
- [CHI 91] CHIAVERINI S., SCIAVICCO L., SICILIANO B., “Control of robotic systems through singularities”, *Proceedings of the International Workshop on Nonlinear and Adaptive Control: Issues in Robotics*, 1990, Lecture Notes in Control and Information Sciences, Grenoble, France, vol. 162, 1991.
- [CRA 89] CRAIG J., *Introduction to Robotics, Mechanics and Control*, Addison-Wesley, Reading, MA, 1989.
- [DOM 04] DOMINGUEZ-LOPEZ J.A., DAMPER R.I., CROWDER R.M., *et al.*, “Adaptive neurofuzzy control of a robotic gripper with on-line machine learning”, *Robotics and Autonomous Systems*, vol. 48, pp. 93–110, 2004.
- [GAR 02] GARCIA E., KELLY R., “Experimental evaluation of task space regulators using unit quaternions on a direct-drive spherical wrist”, *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, DC, May 2002.
- [HOO 91] HOOTSMANS N.A.M., DUBOWSKY S., “Large motion control of mobile manipulators including vehicle suspension characteristics”, *Proceedings of IEEE International Conference on Robotics and Automation*, Sacramento, CA, April 1991.
- [KEL 05] KELLY R., MORENO J., “Manipulator motion control in operational space using joint velocity inner loops”, *Automatica*, vol. 41, no. 8, pp. 1423–1432, 2005.
- [KEY] KEYMASI A., MOOSAVIAN S.A.A., ALI A., “Modified transpose Jacobian for control of a tractor-trailer wheeled robot”, *Proceedings of the RSI/ISM International Conference on Robotics and Mechatronics (ICRoM 2013)*, 12–14 February, Tehran, Iran, forthcoming.
- [MBE 05] MBEDE J.B., ELE P., MVEH-ABIA C.M., *et al.*, “Intelligent mobile manipulator navigation using adaptive neurofuzzy systems”, *Information Sciences*, vol. 171, pp. 447–474, 2005.

- [MEG 05] MEGHDARI A., NADERI D., ALAM M.R., “Neural-network-based observer for real-time tipover estimation”, *Mechatronics*, vol. 15, no. 8, pp. 989–1004, 2005.
- [MIY 88] MIYAZAKI F., MASUTANI Y., ARIMOTO S., “Sensor feedback using approximate Jacobian”, *Proceedings of the USA–Japan Symposium on Flexible Automation: Crossing Bridges, Advances in Flexible Automation and Robotics*, Minneapolis, MN, 18–20 July 1988.
- [MOO 97] MOOSAVIAN S.A.A., PAPADOPOULOS E., “Control of space free-flyers using modified transpose Jacobian algorithm”, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 8–13 September, Grenoble, France, 1997.
- [MOO 04] MOOSAVIAN S.A.A., HOMAEINEJAD M.R., “Regulated sliding mode control of satellite rotation”, *Proceedings of the IFAC Workshop on Generalized Solutions in Control Problems (GSCP-2004)*, 22–26 September, Russia, 2004.
- [MOO 05] MOOSAVIAN S.A.A., RASTEGARI R., *et al.*, “Multiple impedance control for space free-flying robots”, *AIAA Journal of Guidance, Control, and Dynamics*, vol. 28, no. 5, pp. 939–947, September 2005.
- [MOO 07] MOOSAVIAN S.A.A., PAPADOPOULOS E., “Modified transpose Jacobian control of robotic systems”, *Automatica*, vol. 43, no. 7, pp. 1226–1233, 2007.
- [MOO 08] MOOSAVIAN S.A.A., ASHTIANI H.R., “Cooperation of robotic manipulators using non-model-based multiple impedance control”, *Journal of Industrial Robot*, vol. 35, no. 6, pp. 549–558, October 2008.
- [MOO 10] MOOSAVIAN S.A.A., PAPADOPOULOS E., “Cooperative object manipulation with contact impact using multiple impedance control”, *International Journal of Control, Automation, and Systems*, vol. 8, no. 2, pp. 314–327, March 2010.
- [PAP 95] PAPADOPOULOS E., MOOSAVIAN S.A.A., “Dynamics and control of space free-flyers with multiple arms”, *Journal of Advanced Robotics*, vol. 9, no. 6, pp. 603–624, 1995.
- [RAS 10] RASTEGARI R., MOOSAVIAN S.A.A., “Multiple impedance control of space free-flying robots via virtual linkages”, *Acta Astronautica*, vol. 66, pp. 748–759, 2010.
- [SLO 87] SLOTINE J.J.E., LI W., “Adaptive robot control: a new perspective”, *Proceedings of the 26th IEEE Conference on Decision and Control*, Los Angeles, CA, 9–11 December 1987.
- [STE 04] STEIL J.J., RÖTHLING F., HASCHKE R., *et al.*, “Situating robot learning for multi-modal instruction and imitation of grasping”, *Robotics and Autonomous Systems*, vol. 47, pp. 129–141, 2004.

- [TAI 00] TAIRA Y., SAGARA S., KATOH R., “Digital adaptive control of space robot manipulators using transpose of generalized Jacobian matrix”, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 31 October–5 November, Takamatsu, Japan, 2000.
- [UME 89] UMETANI Y., YOSHIDA K., “Resolved motion rate control of space manipulators with generalized Jacobian matrix”, *IEEE Transactions on Robotics and Automation*, vol. 5, no. 3, pp. 303–314, 1989.
- [YOU 87] YUCEF-TOUMI K., ITO O., “Controller design for systems with unknown dynamics”, *Proceedings of American Control Conference*, Minneapolis, MN, 10–12 June 1987.

Chapter 12

Derivative-Free Nonlinear Kalman Filtering for PMSG Sensorless Control

12.1. Introduction

State estimation of nonlinear electric power generators using filters is a significant topic in the area of mechatronics because it can provide improved methods for sensorless control and fault diagnosis of such electromechanical systems. In the design of nonlinear controllers for power generators, it is important to measure all state variables needed to generate a feedback control law. In case of state variables for which direct measurement is technically difficult or costly, estimation can be performed with nonlinear filtering methods [LIN 10, JIA 01, DEL 01, ANE 07, MAH 12]. This chapter proposes derivative-free nonlinear Kalman filtering for sensorless control of permanent magnet synchronous generators (PMSGs).

For nonlinear electric power generators, and under Gaussian noise, the extended Kalman filter (EKF) is frequently applied for estimating the non-measurable state variables through the processing of input and output sequences [KAM 99, BAS 93, HAR 02, NG 03]. The EKF is based on linearization of the dynamical system using Taylor series expansion [XIO 08, RIG 07, RIG 01, RIG 08, RIG 09]. Although EKF is efficient in several estimation problems, it is characterized by cumulative errors due to the

local linearization assumption and this may affect the accuracy of the state estimation or even risk the stability of the observer-based control loop.

It is also known that we can attempt transformation of the PMSG model into the canonical (Brunovsky) form through the application of the Lie-algebra theory. By using such differential geometric methods, it is possible to arrive at a description of the system in the linear canonical form if the relative degree of the system is equal to the order of the system. After transformation to the canonical form, state estimation with the use of a linear system is possible. However, this linearization procedure requires the computation of Lie derivatives (partial derivatives on the vector fields describing the system dynamics), which can be a cumbersome computation procedure.

Aiming at finding more efficient state estimation methods for the model of the PMSG (with reference to the Lie algebra-based state estimator design), in this chapter a derivative-free approach to Kalman filtering is introduced. In the proposed derivative-free Kalman filtering method, the system is first subject to a linearization transformation that is based on the differential flatness theory and then state estimation is performed by applying the standard Kalman filter recursion to the linearized model. Unlike the Lie algebra-based estimator design method, the proposed approach provides estimates of the state vector of the PMSG without the need for derivatives and Jacobians calculation. By avoiding linearization approximations, the proposed filtering method improves the accuracy of estimation of the system state variables, and results in smooth control signal variations and in minimization of the tracking error of the associated control loop [RIG 10, RIG 11a, RIG 11b].

Differential flatness theory is currently a main topic in nonlinear dynamical systems [RUD 03, SIR 04, RIG 11c]. To find out if a dynamical system is differentially flat, the following should be examined: (1) the existence of the so-called flat output, that is a new variable which is expressed as a function of the system's state variables. The flat output and its derivatives should not be coupled in the form of an ordinary differential equation (ODE); (2) the components of the system (that is state variables and control input) should be expressed as functions of the flat output and its derivatives [LEV 11, FLI 99, LAR 07]. In certain cases, the differential flatness theory enables transformation to a linearized form (canonical Brunovsky form) for which the design of the controller becomes easier. In other cases, by showing that a system is differentially flat, we can easily design a reference trajectory as a function of the so-called flat output and can find a control law that assures tracking of this desirable trajectory [FLI 99, VIL 07].

This chapter analyzes differential flatness of the PMSG model and its resulting description in the Brunovsky (canonical) form [MAR 99]. At a second stage and with the use of the linearized model, Kalman filtering is proposed for estimating the non-directly measurable elements of the state vector of the power generator. To enable efficient operation of the power generator under variable mechanical input power (torque), the Kalman filter is extended towards a disturbances estimator. This enables us to estimate simultaneously both the state vector of the generator and the unknown mechanical input power (torque). Finally, state estimation-based control is applied to assure that the rotation speed of the generator tracks a predefined reference set point. The aggregate control input is generated by including in the state-feedback control law a term that is based on the estimation of the unknown input power and of its derivatives and which compensates for the effects of variation of this input power.

The structure of the chapter is as follows: in section 12.2, the dynamic model of the PMSG is analyzed. In section 12.3, it is shown how with the application of Lie algebra theory we can obtain a description of the generator's dynamics in the linear canonical form and based on such a linearization procedure we can design a state estimator for the system. In section 12.4, it is shown how with the use of the differential flatness theory and without the computation of partial derivatives and Jacobian matrices we can obtain again a description of a nonlinear dynamical system in the linear canonical form. In section 12.5, it is shown that the PMSG is a differentially flat system and its nonlinear model can be transformed into an equivalent linear canonical form. Through this linearization, it becomes possible to apply derivative-free nonlinear Kalman filtering for obtaining an estimate of the non-measurable elements of the PMSG's state vector. In section 12.6, methods for robust state estimation based on unknown input observers are analyzed. In section 12.7, it is shown how the derivative-free nonlinear Kalman filter can be generalized in the form of a disturbances estimator, thus enabling simultaneous estimation of both the non-measurable elements of the generator's state vector and the unknown input power (torque). In section 12.8, the proposed state estimation-based control scheme for the power generator is evaluated with the use of simulation experiments. Finally, in section 12.9, concluding remarks are stated.

12.2. Dynamic model of the permanent magnet synchronous generator

It is considered that the third-order model PMSG is connected to the power grid as shown in Figure 12.1, thus forming the model of a single machine infinite bus (SMIB) system.

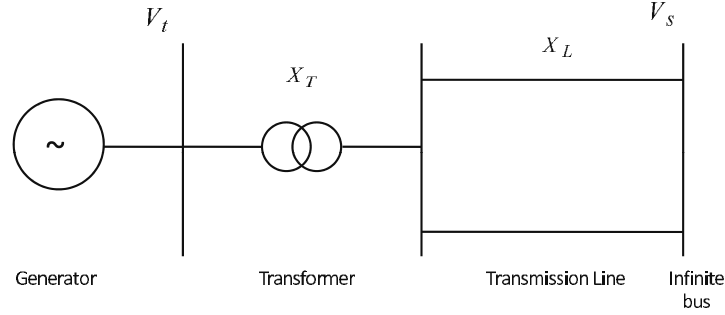


Figure 12.1. PMSG connected to the power grid according to the single machine infinite bus (SMIB) model

The PMSG mechanical dynamics can be represented as follows:

$$\begin{aligned}\dot{\delta} &= \omega - \omega_0 \\ \dot{\omega} &= -\frac{D}{2J}(\omega - \omega_0) + \frac{\omega_0}{2J}(P_m - P_e)\end{aligned}\quad [12.1]$$

where δ is the turn angle of the generator's rotor, ω is the rotation speed of the rotor with respect to synchronous reference, ω_0 is the synchronous speed of the generator, J is the moment of inertia of the rotor, P_m is the mechanical input torque to the generator that is associated with the mechanical input power, D is the damping constant of the generator and T_e is the electrical torque that is associated with the generated active power. Moreover, the following variables are defined: $\Delta\delta = \delta - \delta_0$ and $\Delta\omega = \omega - \omega_0$ with ω_0 denoting the synchronous speed.

The generator's electric dynamics is described by [LIN 10, JIA 01, DEL 01, ANE 07] and [MAH 12]:

$$\dot{E}'_q = \frac{1}{T_{d_o}}(E_f - E_q)\quad [12.2]$$

where E'_q is the quadrature-axis transient voltage of the generator, E_q is the quadrature axis voltage of the generator, T_{d_o} is the direct axis open-circuit transient time constant of the generator and E_f is the equivalent voltage in the excitation coil.

The algebraic equations of the synchronous generator are given by

$$\begin{aligned}
 E_q &= \frac{x_{d\Sigma}}{x'_{d\Sigma}} E'_q - (x_d - x'_d) \frac{V_s}{x'_{d\Sigma}} \cos(\Delta\delta) \\
 I_q &= \frac{V_s}{x'_{d\Sigma}} \sin(\Delta\delta) \\
 I_d &= \frac{E'_q}{x'_{d\Sigma}} - \frac{V_s}{x'_{d\Sigma}} \cos(\Delta\delta) \\
 P_e &= \frac{V_s E'_q}{x'_{d\Sigma}} \sin(\Delta\delta) \\
 Q_e &= \frac{V_s E'_q}{x'_{d\Sigma}} \cos(\Delta\delta) - \frac{V_s^2}{x_{d\Sigma}} \\
 V_t &= \sqrt{(E'_q - X'_d I_d)^2 + (X'_d I_q)^2}
 \end{aligned} \tag{12.3}$$

where $x_{d\Sigma} = x_d + x_T + x_L$, $x'_{d\Sigma} = x'_d + x_T + x_L$, x_d is the direct-axis synchronous reactance, x_T is the reactance of the transformer, x_L is the reactance of the transmission line, I_d and I_q are direct and quadrature axis currents of the generator, V_s is the infinite bus voltage, Q_e is the generator reactive power delivered to the infinite bus, and V_t is the terminal voltage of the generator.

Substituting the electrical equations of the PMSG given in equation [12.3] into the equations of the electrical and mechanical dynamics of the rotor given in equations [12.1] and [12.2], respectively, the complete model of the SMIB model is obtained:

$$\begin{aligned}
 \dot{\delta} &= \omega - \omega_0 \\
 \dot{\omega} &= -\frac{D}{2J}(\omega - \omega_0) + \omega_0 \frac{P_m}{2J} - \omega_0 \frac{1}{2J} \frac{V_s E'_q}{x'_{d\Sigma}} \sin(\Delta\delta) \\
 \dot{E}'_q &= -\frac{1}{T'_d} E'_q + \frac{1}{T_{d_o}} \frac{x_d - x'_d}{x'_{d\Sigma}} V_s \cos(\Delta\delta) + \frac{1}{T_{d_o}} E_f
 \end{aligned} \tag{12.4}$$

where $T'_d = \frac{x_{d\Sigma}}{x'_{d\Sigma}} T_{d_o}$ is the time constant of the field winding.

The previously analyzed SMIB model of the PMSG is described by a nonlinear state space model of the form

$$\dot{x} = f(x) + g(x)u \quad [12.5]$$

where the state vector x is defined as

$$x = (\Delta\delta \ \Delta\omega \ E'_q)^T \quad [12.6]$$

The vector fields $f(x)$ and $g(x)$ are defined as

$$f(x) = \begin{pmatrix} \omega - \omega_0 \\ -\frac{D}{2J}(\omega - \omega_0) + \omega_0 \frac{P_m}{2H} - \omega_0 \frac{1}{2J} \frac{V_s E'_q}{x'_{d\sigma}} \sin(\Delta\delta) \\ -\frac{1}{T'_d} E'_q + \frac{1}{T_{do}} \frac{x_d - x'_d}{x'_{d\sigma}} V_s \cos(\Delta\delta) \end{pmatrix} \quad [12.7]$$

$$g(x) = \begin{pmatrix} 0 & 0 & \frac{1}{T_{do}} \end{pmatrix}^T \quad [12.8]$$

with control input $u = E_f$, the field voltage (equivalent voltage in the excitation coil) and measurable output the rotation speed of the rotor

$$y = h(x) = \delta - \delta_0 \quad [12.9]$$

12.3. Lie algebra-based design of nonlinear state estimators

12.3.1. Relative degree for nonlinear systems

The nonlinear model of the PMSG given in equation [12.4] is in an affine in-the-input form, that is

$$\begin{aligned} \dot{x}(t) &= f(x) + g(x)u \\ y(t) &= h(x) \end{aligned} \quad [12.10]$$

where $x \in R^n$ is the state vector, $u \in R$ is the control vector, $y \in R$ is the output vector, $f(x)$ and $g(x)$ vector fields that belong in R^n and $h(x)$ is the scalar function of x . It will be shown that a state estimator for the nonlinear model of the PMSG can be derived using Lie algebra, which introduces a change of

coordinates (diffeomorphism) that enables us to write a nonlinear system into an equivalent linear form. Next, the following definitions from the Lie algebra theory are used [KHA 96]:

Lie derivative: For a given differentiable scalar function $h(x)$ of $x = [x_1, x_2, x_3, \dots, x_n]^T$ and a vector field $f(x) = [f_1, f_2, f_3, \dots, f_n]^T$, the Lie derivative of function $h(x)$ along the vector field $f(x)$ is a new scalar function defined by $L_f h(x)$ which is obtained as follows:

$$L_f h(x) = \frac{\partial h(x)}{\partial x} f(x) = \sum_{i=1}^n \frac{\partial h(x)}{\partial x_i} f_i(x) \quad [12.11]$$

Relative degree: If the Lie derivative of the function $L_f^{r-1} h(x)$ along vector field $g(x)$ is not equal to zero in a neighborhood Ω , that is

$$L_g L_f^{r-1} h(x) \neq 0 \quad [12.12]$$

then it is said that the system has relative degree r in Ω .

The relative degree of the system is a parameter to take into account in the design of controllers or observers for nonlinear dynamical systems. If the relative degree of a system is equal to its order n , then the system is exactly linearizable. If $r < n$, then the system is partially linearizable.

12.3.2. Nonlinear observer design for exactly linearizable systems

Under the condition $r = n$, we have [KHA 96]:

$$\begin{aligned} L_g L_f^{1-1} h(x) &= L_g L_f^{2-1} h(x) = \dots = L_g L_f^{n-2} h(x) = 0 \\ L_g L_f^{n-1} h(x) &\neq 0 \end{aligned} \quad [12.13]$$

Next, a change of coordinates is performed as follows:

$$z_1 = y = h(x) = L_f^{1-1} h(x) \quad [12.14]$$

Moreover, it holds

$$\dot{z}_1 = \frac{\partial h(x)}{\partial x} \dot{x} \quad [12.15]$$

Substituting the system's dynamics into equation [12.15], we get

$$\begin{aligned}\dot{z}_1 &= \frac{\partial h(x)}{\partial x} f(x) + \frac{\partial h(x)}{\partial x} g(x) \cdot u \\ &= L_f^2 h(x) + L_g L_f^{1-1} h(x) u\end{aligned}\quad [12.16]$$

According to the condition about the relative degree of the system, it holds $L_g L_f^{1-1} h(x) = 0$ and from equation [12.16], we get

$$\dot{z}_1 = L_f h(x) = z_2 \quad [12.17]$$

In a similar way

$$\begin{aligned}\dot{z}_2 &= L_f^2 h(x) = z_3 \\ &\vdots \\ &\vdots \\ \dot{z}_{n-1} &= L_f^{n-1} h(x) = z_n\end{aligned}\quad [12.18]$$

Again, using the property of the system's relative degree that we have $L_g L_f^{n-1} h(x) \neq 0$, we obtain

$$\dot{z}_n = L_f^n h(x) + L_g L_f^{n-1} h(x) u = \alpha(x) + b(x) u = v \quad [12.19]$$

where $\alpha(x) = L_f^n h(x)$, $b(x) = L_g L_f^{n-1} h(x)$ and $v = \alpha(x) + b(x) u$. After this change of coordinates, the system can be written as

$$\dot{z} = Az + Bu \quad [12.20]$$

where

$$z = \phi(x) = \begin{pmatrix} h(x) \\ L_f h(x) \\ \vdots \\ \vdots \\ L_f^{n-1} h(x) \end{pmatrix} \quad [12.21]$$

while matrices A, B appearing in the previous state-space equation are given by

$$A = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \quad [12.22]$$

$$B = (0 \ 0 \ \cdots \ 0 \ 1)^T$$

The state observer for the transformed system of equation [12.20] is

$$\begin{aligned} \dot{\hat{z}} &= A\hat{z} + Bv + K(y - h(\hat{x})) \text{ or} \\ \dot{\hat{z}} &= A\hat{z} + B[\alpha(\hat{x}) + b(\hat{x})u] + K(y - h(\hat{x})) \text{ or} \\ \dot{\hat{z}} &= A\hat{z} + B[\alpha(\phi^{-1}(\hat{z})) + b(\phi^{-1}(\hat{z}))u] + K(y - h(\phi^{-1}(\hat{z}))) \end{aligned} \quad [12.23]$$

It is also possible to express the state observer using a nonlinear model, that is

$$\dot{\hat{x}} = f(\hat{x}) + g(\hat{x})u + L(y - h(\hat{x})) \quad [12.24]$$

It will be shown that the nonlinear observer's gain L is now given by

$$L = (J_\phi(\hat{x}))^{-1}K \quad [12.25]$$

where matrix $J_\phi(\hat{x})$ is the Jacobian of the new coordinates $z_1, \dots, z_n$, which is obtained after the nonlinear change of coordinates, and K is the observer's gain computed for the linearized equivalent of the system of equation [12.23]. The observer gain K for the linearized system can be obtained through the Kalman filter recursion. We have that

$$d\hat{z} = \begin{pmatrix} dh(\hat{x}) \\ dL_f h(\hat{x}) \\ \vdots \\ dL_f^{n-1} h(\hat{x}) \end{pmatrix} = J_\phi(\hat{x})d\hat{x} \quad [12.26]$$

or equivalently

$$\dot{\hat{z}} = J_\phi(\hat{x})\dot{\hat{x}} \quad [12.27]$$

It holds that the Jacobian matrix of $\phi(\hat{x})$ with respect to $\hat{x}$ can be written as

$$J_\phi(\hat{x}) = \frac{\partial \phi(\hat{x})}{\partial \hat{x}} \quad [12.28]$$

Using the state observer dynamics described in equation [12.24], we have

$$\frac{\partial \phi(\hat{x})}{\partial \hat{x}} \dot{\hat{x}} = \frac{\partial \phi(\hat{x})}{\partial \hat{x}} f(\hat{x}) + \frac{\partial \phi(\hat{x})}{\partial \hat{x}} g(\hat{x})u + \frac{\partial \phi(\hat{x})}{\partial \hat{x}} L(\hat{x})(y - h(\hat{x})) \quad [12.29]$$

Considering that for the first row of the Jacobian matrix it holds $\phi(\hat{x}) = h(\hat{x})$, we have

$$\frac{\partial h(\hat{x})}{\partial \hat{x}} \dot{\hat{x}} = \frac{\partial h(\hat{x})}{\partial \hat{x}} f(\hat{x}) + \frac{\partial h(\hat{x})}{\partial \hat{x}} g(\hat{x})u + K(y - h(\hat{x})) \quad [12.30]$$

or equivalently

$$\dot{\hat{z}}_1 = L_f h(\hat{x}) + L_g h(\hat{x})u + K_1(y - h(\hat{x})) \quad [12.31]$$

Moreover, it holds $L_g h(\hat{x}) = 0$, and $L_f h(\hat{x}) = \hat{z}_2$, and thus we obtain

$$\dot{\hat{z}}_1 = \hat{z}_2 + K_1(y - h(\hat{x})) \quad [12.32]$$

In a similar manner, we have

$$\begin{aligned} \dot{\hat{z}}_2 &= \hat{z}_3 + K_2(y - h(\hat{x})) \\ &\dots \\ \dot{\hat{z}}_{n-1} &= \hat{z}_n + K_{n-1}(y - h(\hat{x})) \end{aligned} \quad [12.33]$$

Additionally, using that $L_g L_f^{n-1} \neq 0$, we can finally write

$$\dot{\hat{z}}_n = \alpha(\phi^{-1}(\hat{z})) + b(\phi^{-1}(\hat{z}))u + K_n(y - h(\hat{x})) \quad [12.34]$$

where $\alpha(\phi^{-1}(\hat{z})) = L_f^{n-1} h(\hat{x}) = L_f^{n-1} h(\phi^{-1}(\hat{z}))$, and $b(\phi^{-1}(\hat{z})) = L_g L_f^{n-1} h(\hat{x}) = L_g L_f^{n-1} h(\phi^{-1}(\hat{z}))$. Using the previous notation, we obtain the formulation of the nonlinear estimator's gain L as a function of the observation gain K for the linearized equivalent of the system described in equation [12.23]. Thus, we finally get equation [12.25]

$$(J_\phi(\hat{x}))L = K \Rightarrow L = (J_\phi(\hat{x}))^{-1}K \quad [12.35]$$

12.3.3. Linearization of PMSG dynamics using Lie Algebra

The nonlinear dynamical model of the PMSG was described in equations [12.5]–[12.9]. The linearization procedure with the use of Lie algebra gives:

$$z_1 = L_f^0 h(x) = h(x) = \Delta\delta \quad [12.36]$$

$$z_2 = L_f^1 h(x) = \left(\frac{\partial h}{\partial x_1} \frac{\partial h}{\partial x_2} \frac{\partial h}{\partial x_3} \right) \cdot \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} \Rightarrow \quad [12.37]$$

$$z_2 = x_2 = \Delta\omega$$

$$z_3 = L_f^2 h(x) = \left(\frac{\partial z_2}{\partial x_1} \frac{\partial z_2}{\partial x_2} \frac{\partial z_2}{\partial x_3} \right) \cdot \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} \Rightarrow \quad [12.38]$$

$$z_3 = f_2 = -\frac{D}{2J}(\omega - \omega_0) + \omega_0 \frac{P_m}{2J} - \frac{\omega_0}{2J} \frac{V_s E_q'}{x'_{d\Sigma}} \sin(\Delta\delta)$$

Moreover, according to the previous analysis, it holds

$$\dot{z}_3 = L_f^3 h(x) + L_g L_f^2 h(x) u \quad [12.39]$$

where

$$\begin{aligned} L_f^3 h(x) &= \left(\frac{\partial z_3}{\partial x_1} \frac{\partial z_3}{\partial x_2} \frac{\partial z_3}{\partial x_3} \right) \cdot \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} \Rightarrow \\ L_f^3 h(x) &= \left(-\frac{\omega_0}{2J} \frac{V_s}{x'_{d\Sigma}} x_3 \cos(x_1) \dot{x}_1 - \frac{D}{2J} \left(\frac{-D}{2J} x_2 + \omega_0 \frac{P_m}{2J} \right. \right. \\ &\quad \left. \left. - \frac{\omega_0}{2J} \frac{V_s}{x'_{d\Sigma}} x_3 \sin(x_1) \right) \left(\frac{1}{T_d'} x_3 + \frac{1}{T_{do}} \frac{x_d - x_d'}{x'_{d\Sigma}} \cos(x_1) \right) \right) \Rightarrow [12.40] \\ L_f^3 h(x) &= \left(\frac{D}{2J} \right)^2 \dot{x}_1 - \omega_0 \frac{D}{2J} \frac{P_m}{2J} + \frac{D}{(2J)^2} \omega_0 \frac{V_s}{x'_{d\Sigma}} x_3 \sin(x_1) \\ &\quad + \frac{\omega_0}{2J} \frac{V_s}{x'_{d\Sigma}} \frac{1}{T_d'} x_3 \sin(x_1) - \frac{\omega_0}{2J} \frac{V_s}{x'_{d\Sigma}} \frac{1}{T_{do}} \frac{x_d - x_d'}{x'_{d\Sigma}} V_s \cos(x_1) \sin(x_1) \\ &\quad - \frac{\omega_0}{2H} \frac{V_s}{x'_{d\Sigma}} x_3 \cos(x_1) x_2 \end{aligned}$$

Finally, in an equivalent manner, we obtain

$$\begin{aligned} L_g L_f^2 h(x) &= \left(\frac{\partial z_3}{\partial x_1} \frac{\partial z_3}{\partial x_2} \frac{\partial z_3}{\partial x_3} \right) \cdot \begin{pmatrix} g_1 \\ g_2 \\ g_3 \end{pmatrix} \Rightarrow \\ L_g L_f^2 h(x) &= -\frac{\omega_0}{2J} \frac{1}{T_{do}} \frac{V_s}{x'_{d\Sigma}} \sin(x_1) \end{aligned} \quad [12.41]$$

Defining the control input for the linearized system $v = L_f^3 h(x) + L_g L_f^2 h(x)u$, the state-space model can be written in the following linear canonical (Brunovsky) form:

$$\begin{pmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} v \quad [12.42]$$

12.4. Differential flatness for nonlinear dynamical systems

12.4.1. Definition of differentially flat systems

Differential flatness is a structural property of a class of nonlinear systems, denoting that all system variables (such as state vector elements and control inputs) can be written in terms of a set of specific variables (the so-called flat outputs) and their derivatives. The following nonlinear system is considered:

$$\dot{x}(t) = f(x(t), u(t)) \quad [12.43]$$

The time is $t \in R$, the state vector is $x(t) \in R^n$ with initial conditions $x(0) = x_0$ and the input is $u(t) \in R^m$. Next, the properties of differentially flat systems are given [RUD 03, SIR 04, RIG 11c, LEV 11, FLI 99, LAR 07, VIL 07]:

The finite dimensional system of equation [12.43] can be written in the general form of an ODE, that is $S_i(w, \dot{w}, \ddot{w}, \dots, w^{(i)})$, $i = 1, 2, \dots, q$. The term w is a generic notation for the system variables (these variables are, for instance, the elements of the system's state vector $x(t)$ and the elements of the control input $u(t)$) while $w^{(i)}$, $i = 1, 2, \dots, q$ are the associated derivatives. Such a system is differentially flat if there are m functions $y = (y_1, \dots, y_m)$ of the system variables and of their time derivatives, that is $y_i = \phi(w, \dot{w}, \ddot{w}, \dots, w^{(\alpha_i)})$, $i = 1, \dots, m$ satisfying the following two conditions [RIG 08, FLI 99, MAR 99]:

1) There does not exist any differential relation of the form $R(y, \dot{y}, \dots, y^{(\beta)}) = 0$ which implies that the derivatives of the flat output are not coupled in the sense of an ODE, or equivalently, it can be said that the flat output is differentially independent.

2) All system variables (that is the elements of the system's state vector w and the control input) can be expressed using only the flat output y and its time derivatives $w_i = \psi_i(y, \dot{y}, \dots, y^{(\gamma_i)})$, $i = 1, \dots, s$. An equivalent definition of differentially flat systems is as follows:

DEFINITION.— *The system $\dot{x} = f(x, u)$, $x \in R^n$, $u \in R^m$ is differentially flat if there exist relations*

$$\begin{aligned} h: R^n \times (R^m)^{r+1} &\rightarrow R^m, \\ \phi: (R^m)^r &\rightarrow R^n \text{ and} \\ \psi: (R^m)^{r+1} &\rightarrow R^m \end{aligned} \quad [12.44]$$

such that

$$\begin{aligned} y &= h(x, u, \dot{u}, \dots, u^{(r)}), \\ x &= \phi(y, \dot{y}, \dots, y^{(r-1)}), \text{ and} \\ u &= \psi(y, \dot{y}, \dots, y^{(r-1)}, y^{(r)}). \end{aligned} \quad [12.45]$$

This means that all system dynamics can be expressed as a function of the flat output and its derivatives; therefore, the state vector and the control input can be written as

$$\begin{aligned} x(t) &= \phi(y(t), \dot{y}(t), \dots, y^{(r)}(t)), \text{ and} \\ u(t) &= \psi(y(t), \dot{y}(t), \dots, y^{(r+1)}(t)) \end{aligned} \quad [12.46]$$

It is noted that for linear systems the property of differential flatness is equivalent to that of controllability. Next, an example is given to explain the design of a differentially flat controller for finite dimensional systems of known parameters.

EXAMPLE.— Flatness-based control for a nonlinear system of known parameters. Consider the following model [LAR 07]:

$$\begin{aligned} \dot{x}_1 &= x_3 - x_2 u \\ \dot{x}_2 &= -x_2 + u \\ \dot{x}_3 &= x_2 - x_1 + 2x_2(u - x_2) \end{aligned} \quad [12.47]$$

The candidate flat output is $y_1 = x_1 + \frac{x_2^2}{2}$. Thus, we get:

$$\begin{aligned} y_1 &= x_1 + \frac{x_2^2}{2} \\ y_2 = \dot{y}_1 &= (x_3 - x_2 u) + x_2(u - x_2) = x_3 - x_2^2 \\ y_3 = \dot{y}_2 = \ddot{y}_1 &= x_2 - x_1 + 2x_2(u - x_2) - 2x_2(u - x_2) = -x_1 + x_2 \\ v = \dot{y}_3 = y_1^{(3)} &= -x_3 + x_2 u - x_2 + u = -x_2 - x_3 + u(1 + x_2) \end{aligned} \quad [12.48]$$

It can be verified that property 1 holds, that is there does not exist any differential relation of the form $R(y, \dot{y}, \dots, y^{(\beta)}) = 0$, and this implies that the derivatives of the flat output are not coupled. Moreover, it can be shown that property 2 also holds, that is the components w of the system (elements of the system's state vector and control input) can be expressed using only the flat output y and its time derivatives $w_i = \psi_i(y, \dot{y}, \dots, y^{(\gamma_i)})$, $i = 1, \dots, s$.

For instance, to calculate x_1 with respect to $y_1, \dot{y}_1, \ddot{y}_1$ and $y_1^{(3)}$, the relation of $\ddot{y}_1$ is used, that is:

$$x_1^2 + 2x_1(1 + \ddot{y}_1) + \ddot{y}_1^2 - 2y_1 = 0 \quad [12.49]$$

from which two possible solutions are derived, that is $x_1 = -(1 + \ddot{y}_1 - \sqrt{1 + 2(y_1 + \ddot{y}_1)})$ and $x_1 = -(1 + \ddot{y}_1 + \sqrt{1 + 2(y_1 + \ddot{y}_1)})$. Keeping the biggest of these two solutions, we obtain:

$$\begin{aligned} x_1 &= -(1 + \ddot{y}_1) + \sqrt{1 + 2(y_1 + \ddot{y}_1)} \\ x_2 &= \dot{y}_1 + x_1 \\ x_3 &= \dot{y}_1 + \ddot{y}_1^2 + 2x_1\ddot{y}_1 + x_1^2 \\ u &= \frac{y_1^3 + \ddot{y}_1^2 + \ddot{y}_1 + \dot{y}_1 + x_1 + 2x_1\ddot{y}_1 + x_1^2}{1 + x_1 + \ddot{y}_1} \end{aligned} \quad [12.50]$$

The computation of the equivalent model of the system in the linear canonical form is summarized as follows: by finding the derivatives of the flat output, we get a set of equations that can be solved with respect to the state variables and the control input of the initial state-space description of the system. First, the binomial of variable x_1 given in equation [12.49] is solved providing x_1 as a function of the flat output and its derivatives. Next, using the expression for x_1 and equation [12.50], state variable x_2 is also written as a function of the flat output and its derivatives. Finally, using the expressions for both x_1 and x_2 and equation [12.50], state variable x_3 is written as a function of the flat output and its derivatives. Thus, we can finally express the state vector elements and the control input as a function of the flat output and its derivatives, which completes the proof about differential flatness of the system.

From equation [12.50], it can be concluded that the initial system of equation [12.47] is indeed differentially flat. Using the flat output and its derivatives, the system of equation [12.47] can be written in Brunovsky (canonical) form:

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} v \quad [12.51]$$

Therefore, a transformation of the system into a linear equivalent is obtained and then it is straightforward to design a controller based on linear control theory. Thus, given the reference trajectory $[x_1^*, x_2^*, x_3^*]^T$, we can find the transformed reference trajectory $[y_1^*, \dot{y}_1^*, \ddot{y}_1^*]^T$ and select the appropriate control input v that succeeds tracking. Knowing v , the control u of the initial system can be found.

12.4.2. Classes of differentially flat systems

Main classes of nonlinear differentially flat systems are [MAR 99]:

- 1) Affine in-the-input systems: the dynamics of such systems is given by:

$$\dot{x} = f(x) + \sum_{i=1}^m g_i(x) u_i \quad [12.52]$$

From equation [12.52], we can notice that the above state equation can also describe MIMO dynamical systems. Without loss of generality, it is assumed that $G = [g_1, \dots, g_m]$ is of rank m . In a case where the flat outputs of the aforementioned system are only functions of states x , then this class of dynamical systems is called 0-flat. It has been proved that a dynamical affine system with n states and $n - 1$ inputs is 0-flat if it is controllable.

- 2) Driftless systems: these are systems of the form

$$\dot{x} = \sum_{i=1}^m f_i(x) u_i \quad [12.53]$$

For driftless systems with two inputs, that is

$$\dot{x} = f_1(x) u_1 + f_2(x) u_2 \quad [12.54]$$

flatness holds, if and only if the rank of matrix $E_{k+1} = \{E_k, [E_k, E_k]\}$, $k \geq 0$ (with $E_0 = \{f_1, f_2\}$ and $[E_k, E_k] = \{[X, Y], X, Y \in E_k\}$ denoting a Lie

bracket) is equal to $k + 2$ for $k = 0, \dots, n - 2$. It has been proved that a driftless system, which is differentially flat, is also 0-flat (which means that the flat output is a function of only the elements of the state vector of the initial nonlinear system).

Moreover, for flat systems with n states and $n - 2$ control inputs, that is

$$\dot{x} = \sum_{i=1}^{n-2} u_i f_i(x) \quad x \in R^n \quad [12.55]$$

the flatness property holds if controllability also holds. Furthermore, the system is 0-flat if n is even.

12.4.3. Differential flatness and transformation into the canonical form

The classes of systems for which the application of differential flatness theory results into a canonical form have to be defined. Transformation into the Brunovsky form can be succeeded for systems that admit static feedback linearization (that is a change of coordinates for both the system state variables and the system's control input). Single-input differentially flat systems admit static feedback linearization and therefore can be finally written in the Brunovsky form. For flat multi-input systems, necessary and sufficient conditions that allow an endogenous transformation into Brunovsky coordinates can be also stated [MAR 99, RIG 11].

In particular, for the case of a single-input differentially flat dynamical system, we have:

$$\dot{x} = f_s(x, t) + g_s(x, t)(u + \tilde{d}), \quad x \in R^n, \quad u \in R, \quad \tilde{d} \in R \quad [12.56]$$

where $f_s(x, t)$, $g_s(x, t)$ are nonlinear vector fields that define the system's dynamics, u denotes the control input and $\tilde{d}$ denotes additive input disturbances. Knowing that the system of equation [12.56] is differentially flat, the next step is to try to write it into a Brunovsky form. The selected flat output is again denoted by y . Then, as analyzed in section 12.4, for the state variables x_i of the system of equation [12.56], it holds

$$x_i = \phi_i(y, \dot{y}, \dots, y^{(r-1)}), \quad i = 1, \dots, n \quad [12.57]$$

while for the control input, it holds

$$u = \psi(y, \dot{y}, \dots, y^{(r-1)}, y^{(r)}) \quad [12.58]$$

Introducing the new state variables $y_1 = y$ and $y_i = y^{(i-1)}$, $i = 2, \dots, n$, the initial system of equation [12.56] can be written in the Brunovsky form:

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \vdots \\ \dot{y}_{n-1} \\ \dot{y}_n \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n-1} \\ y_n \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} v \quad [12.59]$$

where $v = f(x, t) + g(x, t)(u + \tilde{d})$ is the control input for the linearized model, and $\tilde{d}$ denotes additive input disturbances.

12.5. Differential flatness of the PMSG

It will be shown that the dynamic model of the PMSG is a differentially flat one, that is it holds that all state variables and its control inputs can be written as functions of the flat output and its derivatives. Moreover, it will be shown that by expressing the elements of the state vector as functions of the flat output and its derivatives, we obtain a transformation of the PMSG model into the linear canonical (Brunovsky) form.

Defining the state vector of the PMSG as $x = [\Delta\delta, \Delta\omega, E'_q]^T$, we have that

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{D}{2J}x_2 + \omega_0 \frac{P_m}{2J} - \frac{\omega_0}{2J} \frac{V_s}{x'_{d\Sigma}} x_3 \sin(x_1) \\ \dot{x}_3 &= -\frac{1}{T_d'} x_3 + \frac{1}{T_{do}} \frac{x_d - x_d'}{x'_{d\Sigma}} V_s \cos(x_1) + \frac{1}{T_{do}} u \end{aligned} \quad [12.60]$$

The flat output is chosen to be $y = x_1$. Therefore, it holds

$$x_1 = y \quad [12.61]$$

$$x_2 = \dot{y} \quad [12.62]$$

while from the second row of equation [12.60], we have

$$\ddot{y} = -\frac{D}{2J}\dot{y} + \omega_0 \frac{P_m}{2J} - \frac{\omega_0}{2J} \frac{V_s x_3}{x'_{d\Sigma}} \sin(y) \quad [12.63]$$

Thus, for $x_1 \neq \pm n\pi$ (where $n = 0, 1, 2, \dots$), we obtain

$$x_3 = \frac{\omega_0 \frac{P_m}{2J} - \ddot{y} - \frac{D}{2J} \dot{y}}{\frac{\omega_0}{2J} \frac{V_s x_3}{x_{d\Sigma}} \sin(y)}, \text{ or} \quad [12.64]$$

$$x_3 = f_a(y, \dot{y}, \ddot{y})$$

From the third row of equation [12.60], we have

$$u = T_{do} [\dot{x}_3 + \frac{1}{T_d} x_3 \frac{1}{T_{do}} \frac{x_d - x_d'}{x_{d\Sigma}'} V_s \cos(x_1)], \text{ or} \quad [12.65]$$

$$u = f_b(y, \dot{y}, \ddot{y})$$

Therefore, all state variables and the control input of the PMSG can be written as functions of the flat output and its derivatives, and the PMSG model is a differentially flat one.

Next, the following change of variables is performed: $y_1 = y$, $y_2 = \dot{y}$, $y_3 = \ddot{y}$. It also holds

$$\dot{y}_3 = y^{(3)} = \ddot{x}_2 \Rightarrow \quad [12.66]$$

$$y^{(3)} = -\frac{D}{2J} \dot{x}_2 - \frac{\omega_0}{2H} \frac{V_s'}{x_{d\Sigma}} \dot{x}_3 \sin(x_1) - \frac{\omega_0}{2J} \frac{V_s'}{x_{d\Sigma}} x_3 \cos(x_1) \dot{x}_1$$

By substituting $\dot{x}_2$ and $\dot{x}_3$ from the second and third rows of equation [12.60], respectively, and after intermediate operations, we obtain

$$y^{(3)} = f_c(y, \dot{y}, \ddot{y}) + g_c(y, \dot{y}, \ddot{y})u \quad [12.67]$$

where

$$f_c(y, \dot{y}, \ddot{y}) = \left(\frac{D}{2J^2}\right) \dot{y} - \omega_0 \frac{D}{2J} \frac{P_m}{2J} + \omega_0 \frac{D}{(2J)^2} \frac{V_s}{x_{d\Sigma}'} x_3 \sin(\dot{y})$$

$$+ \frac{\omega_0}{2J} \frac{V_s}{x_{d\Sigma}'} \frac{1}{T_d} x_3 \sin(y) - \quad [12.68]$$

$$- \frac{\omega_0}{2J} \frac{V_s}{x_{d\Sigma}'} \frac{1}{T_{do}} \frac{x_d - x_d'}{x_{d\Sigma}'} V_s \cos(y) \sin(y) - \frac{\omega_0}{2J} \frac{V_s}{x_{d\Sigma}'} x_3 \cos(y) \dot{y}$$

$$g_c(y, \dot{y}, \ddot{y}) = -\frac{\omega_0}{2J} \frac{1}{T_{do}} \frac{V_s}{x_{d\Sigma}'} \sin(y) \quad [12.69]$$

Thus, the system can be written in the following linear canonical (Brunovsky) form:

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} v \quad [12.70]$$

while the new control input for the linearized system is $v = f_c(y, \dot{y}, \ddot{y}) + g_c(y, \dot{y}, \ddot{y})u$. It can be noticed that the linearized equivalent of the system that is obtained after applying differential flatness theory is the same with the one obtained after applying the Lie algebra-based method.

The controller design for the linearized system described in equation [12.70] is carried out using common pole placement methods. Since the overall system dynamics is described by the differential equation

$$y^{(3)} = v \quad [12.71]$$

a suitable feedback control law that makes the flat output track a desirable set point $y_d(t)$ is given by

$$v = y_d^{(3)}(t) - k_1(\ddot{y}(t) - \ddot{y}_d(t)) - k_2(\dot{y}(t) - \dot{y}_d(t)) - k_3(y(t) - y_d(t)) \quad [12.72]$$

The control input that is finally applied to the PMSG is given by

$$u(t) = g_c^{-1}(t)[v(t) - f_c(t)] \quad [12.73]$$

The control law of equation [12.72] results in the closed-loop dynamics

$$e^{(3)}(t) + k_1\ddot{e}(t) + k_2\dot{e}(t) + k_3e(t) = 0 \quad [12.74]$$

By selecting the feedback control gains k_1, k_2 and k_3 such that the associated characteristic polynomial $p(s) = s^3 + k_1s^2 + k_2s + k_3$ is a Hurwitz one, that is it has poles in the left complex semi-plane, we have that the tracking error converges asymptotically to zero

$$\lim_{t \rightarrow \infty} e(t) = 0 \Rightarrow \lim_{t \rightarrow \infty} y(t) = y_d(t) \quad [12.75]$$

Because the flat output converges to the desirable set point and because all state variables x_1, x_2 and x_3 are expressed as functions of the flat output and its derivatives, we have that the state variables also converge to the desirable set points and therefore efficient control of the PMSG is achieved.

Moreover, for the linearized equivalent of the system, we can perform state estimation with the use of the standard Kalman filter recursion and can compute also the state vector elements that are not directly measurable (i.e. ω and E'_q).

12.6. Robust state estimation-based control of the PMSG

12.6.1. Unknown input observers

The previous method for PMSG controller design was based on the assumption that the mechanical input power (torque) P_m that is applied to the PMSG can be directly measured and that this input torque is a constant or piecewise constant variable. However, this is not always the case since the measurement of P_m may be technically difficult while P_m maybe a time-varying signal (e.g. mechanical excitation due to wind, steam or water flow). In the latter case, we can consider variable P_m as an external disturbance to the generator's model. The simultaneous estimation of the non-measurable elements of the PMSG state vector (that is ω and E'_q) as well as the estimation of the disturbance term (input torque P_m) is possible if using a disturbance estimator is used [COR 05, COR 06, CHE 00, GUP 11, MIK 06].

A first type of disturbance estimator is the unknown input observer. This is applied to dynamical systems of the form

$$\begin{aligned}\dot{x} &= Ax + B(u + w_e) \\ z &= Cx\end{aligned}\tag{12.76}$$

while the disturbance dynamics is given by

$$\begin{aligned}\dot{d} &= A_f d \\ w_e &= C_f d\end{aligned}\tag{12.77}$$

Then, the unknown input observer provides a state estimate of the extended state vector

$$\begin{pmatrix} \hat{\dot{x}} \\ \hat{\dot{d}} \end{pmatrix} = \begin{pmatrix} A & BC_f \\ 0 & A_f \end{pmatrix} \begin{pmatrix} \hat{x} \\ \hat{d} \end{pmatrix} + \begin{pmatrix} B \\ 0 \end{pmatrix} u + K(z - C\hat{x})\tag{12.78}$$

In the generic case, we can assume that the disturbances vector w_e varies dynamically in time. However, in several cases, it suffices to assume a constant or piecewise constant disturbance $\dot{w}_e(z) = 0$, where $A_f = 0$ and $C_f = 1$. The observer's gain can be obtained through the standard Kalman filter recursion.

12.6.2. Perturbation observer

The perturbation observer is an extension of the unknown inputs observer that takes into account not only external disturbances but also parametric

uncertainties. In the discrete-time form, the system dynamics is given by

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + w_f(k) \\ z(k) &= Cx(k) \end{aligned} \quad [12.79]$$

while the disturbance dynamics is given by

$$\begin{aligned} d(k) &= A_f d(k-1) + B_f (B^+ (\hat{x}(k) - A\hat{x}(k-1)) - u(k-1)) \\ \hat{w}_f(k) &= C_f d(k) \\ \hat{x}(k+1) &= A\hat{x}(k) + B(u(k) + \hat{w}_f(k) + L(z(k) - C\hat{x}(k))) \end{aligned} \quad [12.80]$$

where B^+ is the Moore–Penrose pseudo-inverse of matrix B . The unknown input can represent traditional external disturbances and model uncertainties, that is $w_f = w_e + \Delta Ax_k + \Delta Bu_k$.

12.6.3. Extended state observer

The extended state observer uses a canonical form, so the unmodeled dynamics appear at the disturbance estimation part. The system's description in the canonical form is given by

$$\begin{aligned} x_1^{(n)} &= f(x, t, u, w_f) + b_m u \\ z &= x_1 \\ x &= \left(x_1 \ \dot{x}_1 \ \dots \ x_1^{(n-1)} \right)^T \end{aligned} \quad [12.81]$$

$$\begin{pmatrix} \dot{\hat{x}}_1 \\ \vdots \\ \dot{\hat{x}}_{n-1} \\ \dot{\hat{x}}_n \\ \dot{\hat{f}} \end{pmatrix} = \begin{pmatrix} \hat{x}_2 \\ \vdots \\ \hat{x}_n \\ \hat{f} + b_m u \\ 0 \end{pmatrix} + K(x_1 - \hat{x}_1) \quad [12.82]$$

The extended state observer can be also modified to take into account derivatives of the disturbance

$$\begin{aligned} x_1^{(n)} &= f(x, t, u, w_f) + b_m u \\ z &= x_1 \\ x &= \left(x_1 \ \dot{x}_1 \ \dots \ x_1^{(n-1)} \right)^T \\ F &= \left(f \ \dot{f} \ \dots \ f^{(h-1)} \right)^T \end{aligned} \quad [12.83]$$

and now the state and disturbance observer takes the form

$$\begin{pmatrix} \dot{\hat{x}}_1 \\ \dots \\ \dot{\hat{x}}_{n-1} \\ \dot{\hat{x}}_n \\ \dot{\hat{F}}_1 \\ \dots \\ \dot{\hat{F}}_{h-1} \\ \dot{\hat{F}}_h \end{pmatrix} = \begin{pmatrix} \hat{x}_2 \\ \dots \\ \hat{x}_n \\ \hat{f} + b_m u \\ \hat{F}_2 \\ \dots \\ \hat{F}_h \\ 0 \end{pmatrix}^T + K(x_1 - \hat{x}_1) \quad [12.84]$$

The latter form of the extended state observer described in equation [12.84] enables to track various types of disturbances. For example, $h = 1$ allows the estimation of disturbance dynamics defined by its first-order derivative, and $h = 2$ allows the estimation of disturbance dynamics defined by its second-order derivative.

12.7. Estimation of PMSG disturbance input with Kalman filtering

12.7.1. State estimation with the derivative-free nonlinear Kalman filter

It was shown that using differential flatness theory, the initial nonlinear model of the PMSG can be written in the canonical form:

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} v \quad [12.85]$$

Thus, we have a linear model of the form

$$\begin{aligned} \dot{y}_f &= A_f y_f + B_f v \\ z_f &= C_f y_f \end{aligned} \quad [12.86]$$

where $y_f = [y_1, y_2, y_3]^T$ and matrices A_f, B_f, C_f are in the canonical form

$$A_f = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \quad B_f = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad C_f^T = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad [12.87]$$

where the measurable variable $y_1 = \delta$ is associated with the turn angle of the rotor. For the previous model, and after carrying out discretization of matrices A_f , B_f and C_f with common discretization methods, we can perform linear Kalman filtering using equation [12.98] and equation [12.99]. This is *derivative-free nonlinear Kalman filtering* for the model of the generator, which, unlike EKF, is performed without needing the computation of Jacobian matrices and does not introduce numerical errors due to approximative linearization with Taylor series expansion [RIG 11a, RIG 11b, RIG 12a].

12.7.2. Kalman filter-based estimation of disturbances

Up to now, the mechanical input torque of the generator, which has been denoted by P_m , was considered to be constant or piecewise constant while it has been also considered that it is possible to measure it. Now it is assumed that the mechanical input torque varies in time, and in that case the aggregate disturbance input exerted on the generator's model is

$$T_m = -\omega_0 \frac{D}{(2J)^2} P_m + \frac{\omega_0}{2J} \dot{P}_m \quad [12.88]$$

It is also assumed that the dynamics of the disturbance term T_m is defined by its n -th order derivative $T_m^{(n)}$. Considering now that after expressing the system's state variables as functions of the flat outputs and their derivatives, the PMSG's dynamics is given by

$$\begin{aligned} y^{(3)} &= f_c(y, \dot{y}, \ddot{y}) + g_c(y, \dot{y}, \ddot{y})u + \frac{1}{2H} \dot{P}_m \text{ or} \\ y^{(3)} &= v - \omega_0 \frac{D}{(2J)^2} P_m + \frac{\omega_0}{2J} \dot{P}_m \text{ or} \\ y^{(3)} &= v - T_m \end{aligned} \quad [12.89]$$

where

$$\begin{aligned} f_c(y, \dot{y}, \ddot{y}) &= \left(\frac{D}{2J^2} \right) \dot{y} - \omega_0 \frac{D}{2J} \frac{P_m}{2J} + \frac{D}{(2J)^2} \omega_0 \frac{V_s}{x'_{d\Sigma}} x_3 \sin(\dot{y}) \\ &\quad + \frac{\omega_0}{2J} \frac{V_s}{x'_{d\Sigma}} \frac{1}{T_d'} x_3 \sin(y) \\ &\quad - \frac{\omega_0}{2J} \frac{V_s}{x'_{d\Sigma}} \frac{1}{T_{do}} \frac{x_d - x'_d}{x'_{d\Sigma}} V_s \cos(y) \sin(y) - \frac{\omega_0}{2H} \frac{V_s}{x'_{d\Sigma}} x_3 \cos(y) \dot{y} \end{aligned} \quad [12.90]$$

$$g_c(y, \dot{y}, \ddot{y}) = -\frac{D}{2M} \frac{\omega_0}{T_{do}} \frac{V_s}{x'_{d\Sigma}} \sin(y) \quad [12.91]$$

$$\begin{pmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \dot{y}_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \left(v - \omega_0 \frac{D}{(2J)^2} P_m + \frac{\omega_0}{2J} \dot{P}_m \right) \quad [12.92]$$

Next, the state vector of the model of equation [12.92] is extended to include as additional state variables the disturbance input T_m and its derivatives. Then, in the new state-space description, we have $z_1 = y_1$, $z_2 = y_2$, $z_3 = y_3$, $z_4 = T_m = -\omega_0 \frac{D}{(2J)^2} P_m + \frac{\omega_0}{2J} \dot{P}_m$, $z_5 = \dot{T}_m$, and $z_6 = \ddot{T}_m$. Without loss of generality, the disturbance input dynamics is assumed to be described by its third-order derivative $\dot{z}_6 = T_m^{(3)}$. Using the previous definition of state variables, we have the matrix equations

$$\dot{z} = \tilde{A} \cdot z + \tilde{B} \cdot \tilde{v} \quad [12.93]$$

where the control input is

$$\tilde{v} = \begin{pmatrix} v & T_m^{(3)} \end{pmatrix}^T \quad [12.94]$$

with

$$\tilde{A} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \tilde{B} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{pmatrix} \quad \tilde{C}^T = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad [12.95]$$

where the measurable state variable is z_1 . Because the dynamics of the disturbance input are taken to be unknown in the design of the associated disturbances' estimator, we have the following dynamics:

$$\dot{z} = \tilde{A}_o \cdot z + \tilde{B}_o \cdot \tilde{v} + K(C_o z - C_o \hat{z}) \quad [12.96]$$

where $K \in R^{6 \times 1}$ is the state estimator's gain and

$$\tilde{A}_o = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \tilde{B}_o = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \quad \tilde{C}_o^T = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad [12.97]$$

It can be confirmed that the disturbance observer model of the PMSG defined in equation [12.97] is observable. Defining the discrete-time equivalents of matrices $\tilde{A}_o$, $\tilde{B}_o$ and $\tilde{C}_o$ as $\tilde{A}_d$, $\tilde{B}_d$, and $\tilde{C}_d$, respectively, a derivative-free nonlinear Kalman filter can be designed for the aforementioned representation of the system dynamics. The associated Kalman filter-based disturbance estimator is given by [RIG 12].

Measurement update:

$$\begin{aligned} K(k) &= P^-(k) \tilde{C}_d^T [\tilde{C}_d P^-(k) \tilde{C}_d^T + R]^{-1} \\ \hat{z}(k) &= \hat{z}^-(k) + K(k) [z(k) - \tilde{C}_d \hat{x}^-(k)] \\ P(k) &= P^-(k) - K(k) \tilde{C}_d P^-(k) \end{aligned} \quad [12.98]$$

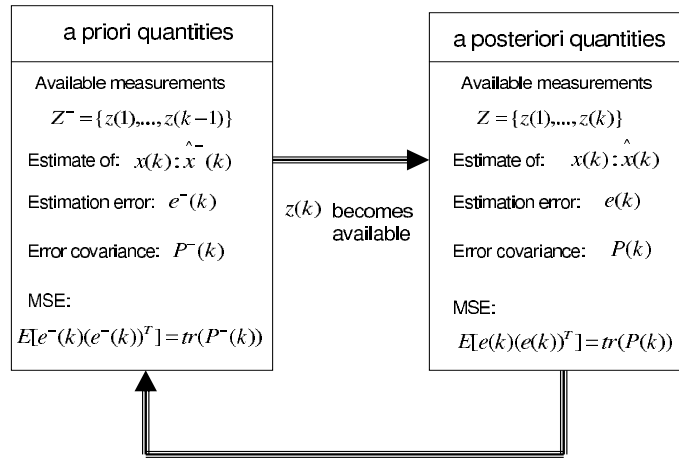


Figure 12.2. Schematic diagram of the Kalman filter loop

Time update:

$$\begin{aligned} P^-(k+1) &= \tilde{A}_d(k) P(k) \tilde{A}_d^T(k) + Q(k) \\ \hat{z}^-(k+1) &= \tilde{A}_d(k) \hat{z}(k) + \tilde{B}_d(k) \tilde{v}(k) \end{aligned} \quad [12.99]$$

To compensate for the disturbance terms, it suffices to use in the control loop the modified control input, which actually removes the effects of the external disturbance variable T_m .

$$v^* = v - \hat{T}_m \quad \text{or} \quad v^* = v - \hat{z}_4 \quad [12.100]$$

12.8. Simulation experiments

To evaluate the performance of the proposed nonlinear control scheme, which uses Kalman filtering to estimate the non-measurable state vector elements of the PMSG and the external disturbances, simulation experiments have been carried out. Different rotation speed set points had been assumed. Moreover, different input torques (mechanical input power profiles) have been considered to affect the PMSG dynamic model (SMIB model). The control loop used in the PMSG control is shown in Figure 12.3.

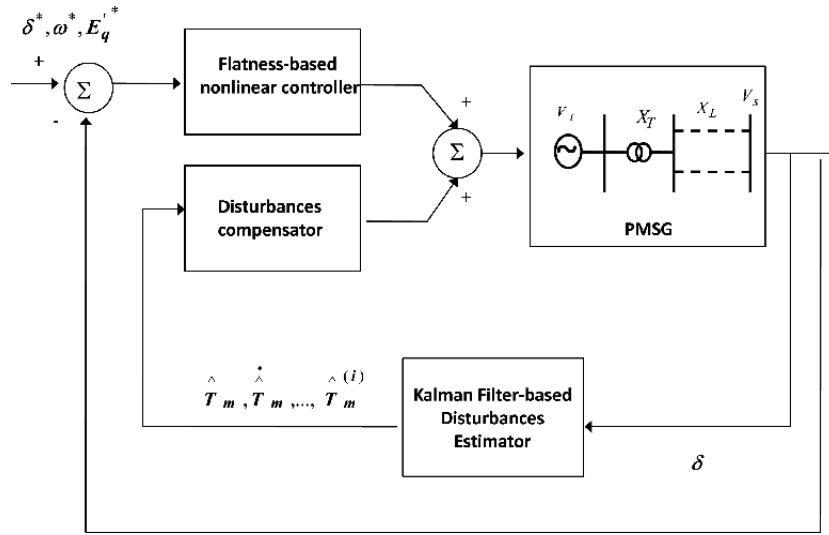


Figure 12.3. Control loop for the PMSG comprising a flatness-based nonlinear controller and a Kalman filter-based disturbances estimator

First, the case of measurable piecewise constant mechanical input power (torque) was examined. It was assumed that the input torque could be measured and could be directly used in the controller's design. The derivative-free nonlinear Kalman filter enabled the estimation of specific elements of the state vector, such as ω and E_q' . The associated results about the tracking performance of the control loop are shown in Figures 12.4 and 12.5. The desirable set point is denoted by a continuous line, the real value of the state variable is denoted by a dashed line and the estimated value of the state variable is shown in dotted line. The units of the PMSG state variables have been expressed in a per unit (p.u.) system. We can observe that the proposed sensorless control scheme for the PMSG succeeds fast and accurate convergence to the desirable set points.

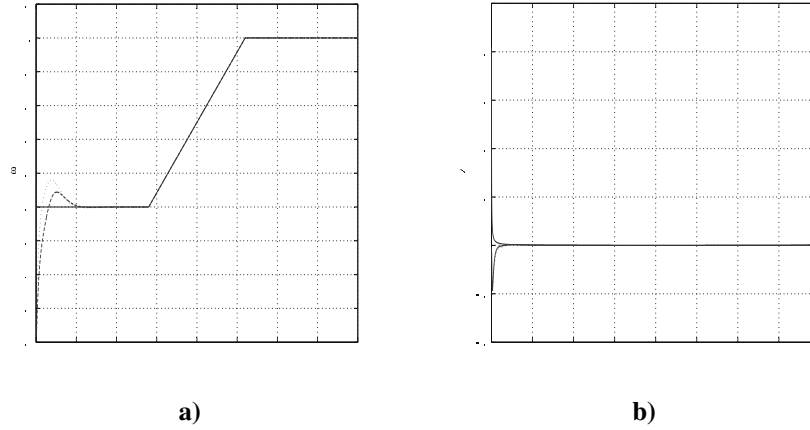


Figure 12.4. Sensorless control of the PMSG under measurable mechanical input torque in the case of speed reference set point 1: a) convergence of the real and estimated values of the angular speed difference $\Delta\omega$ and b) convergence of the real and estimated values of the quadrature axis voltage E'_q

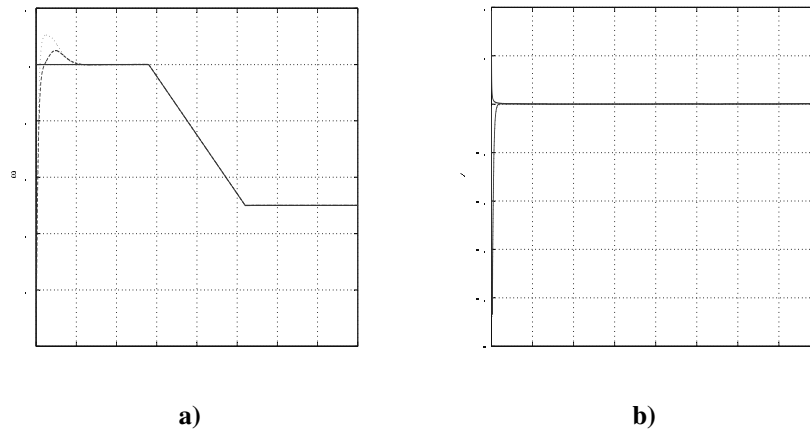


Figure 12.5. Sensorless control of the PMSG under measurable mechanical input torque in the case of speed reference set point 2: a) convergence of the real and estimated values of the angular speed difference $\Delta\omega$ and b) convergence of the real and estimated values of the quadrature axis voltage E'_q

Next, the case of PMSG operation under unknown input power (torque) was examined. The input power was considered to be a disturbance input to the PMSG model and it was assumed that its change in time was defined by the third derivative of the associated variable, that is $T_m^{(3)}$, where $T_m = -\omega_0 \frac{D}{(2J)^2} P_m + \frac{\omega_0}{2J} \dot{P}_m$. The disturbance dynamics was completely unknown to the controller and its identification was performed in real time by the disturbance estimator. It is shown that the derivative-free nonlinear Kalman filter redesigned as a disturbance observer is capable of estimating simultaneously the non-measurable state variables of the generator (ω and E'_q), as well as the unknown disturbance input T_m . A first set of results is concerned with the tracking performance of the control loop in case of piecewise constant disturbance input, as shown in Figures 12.6 and 12.7. The estimation of the piecewise constant disturbance input is shown in Figure 12.8. In addition, results about the tracking performance of the control loop in case of time-varying disturbance input are shown in Figures 12.9 and 12.10, while the estimation of the time-varying disturbance input is shown in Figure 12.11. The units of the PMSG state variables have been expressed again in the p.u. system.

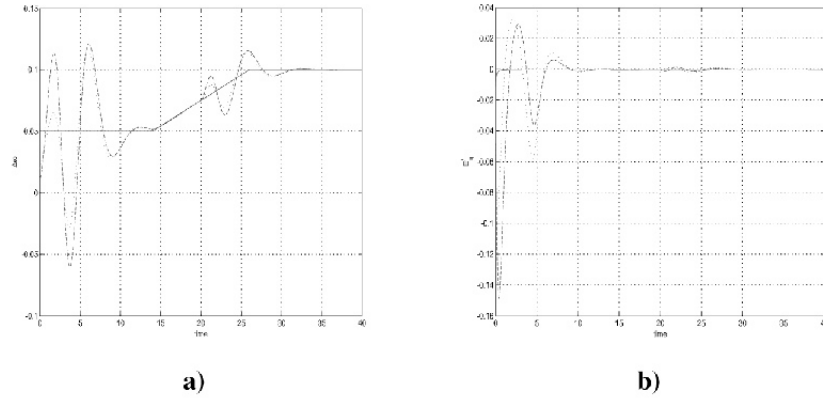


Figure 12.6. Sensorless control of the PMSG under non-measurable (piecewise constant) mechanical input torque in the case of speed reference set point 1: a) convergence of the real and estimated values of the angular speed difference $\Delta\omega$ and b) convergence of the real and estimated values of the quadrature axis voltage E'_q

The simulation experiments have confirmed that the proposed state estimation-based control scheme not only enables implementation of PMSG control through the measurement of a small number of variables (e.g. of only the rotor's turn angle) but also improves the robustness of the PMSG control loop in the case of varying speed set points and varying mechanical input torque.

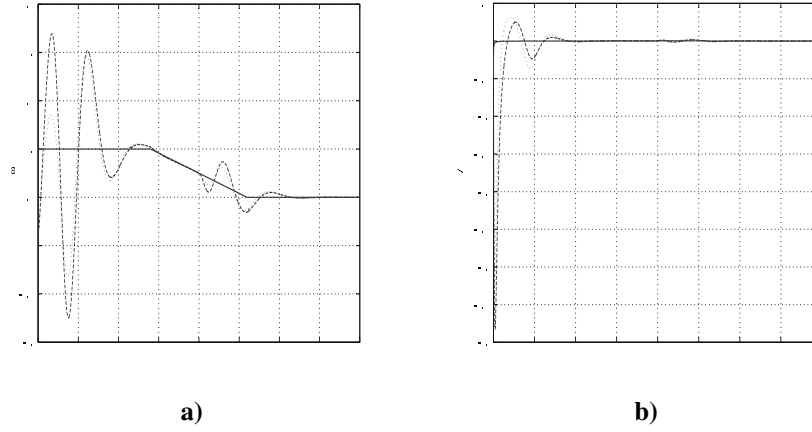


Figure 12.7. Sensorless control of the PMSG under non-measurable (piecewise constant) mechanical input torque in the case of speed reference set point 2: a) convergence of the real and estimated values of the angular speed difference $\Delta\omega'$ and b) convergence of the real and estimated values of the quadrature axis voltage E'_q

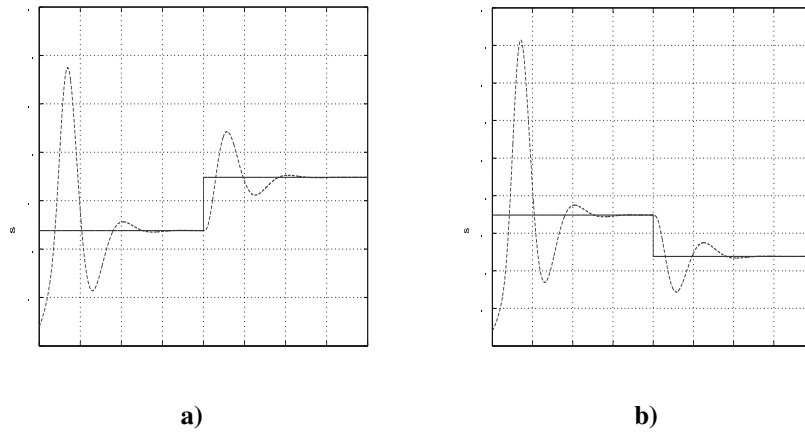


Figure 12.8. Estimation of the non-measurable (piecewise constant) input torque in sensorless control of the PMSG through the processing of rotor angle measurements: a) in the case of speed reference set point 1 and b) in the case of speed reference set point 2

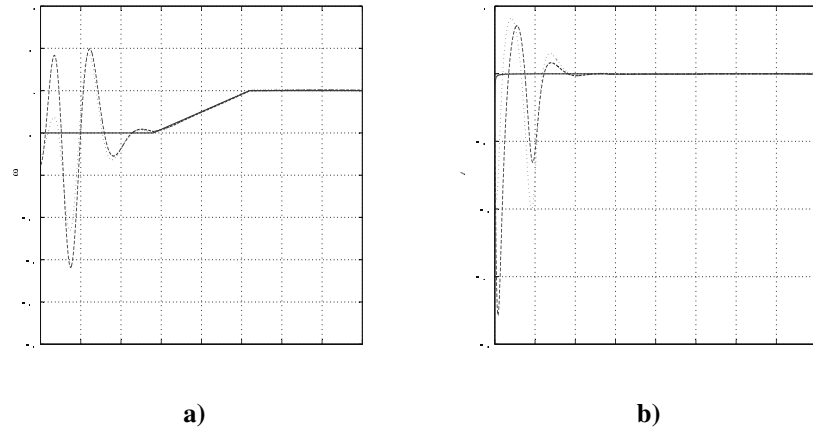


Figure 12.9. Sensorless control of the PMSG under non-measurable (time varying) mechanical input torque in the case of speed reference set point 1: a) convergence of the real and estimated values of the angular speed difference $\Delta\omega$ and b) convergence of the real and estimated values of the quadrature axis voltage E'_q

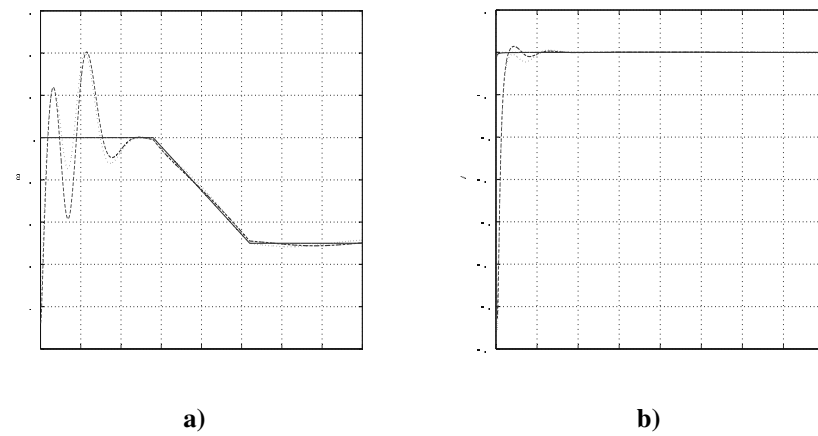


Figure 12.10. Sensorless control of the PMSG under non-measurable (time varying) mechanical input torque in the case of speed reference set point 2: a) convergence of the real and estimated values of the angular speed difference $\Delta\omega$ and b) convergence of the real and estimated values of the quadrature axis voltage E'_q

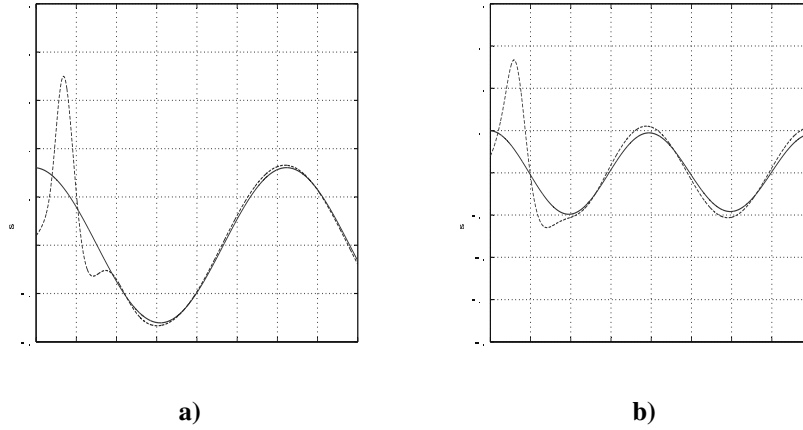


Figure 12.11. Estimation of the non-measurable (time varying) input torque in sensorless control of the PMSG: a) in the case of speed reference set point 1 and b) in the case of speed reference set point 2

12.9. Conclusions

A new method for state estimation-based (sensorless) control of the PMSG is proposed. The method makes use of the differential flatness theory that enables us to transform the initial nonlinear model of the generator into a linear canonical (Brunovsky) form. To show that the PMSG model is a differentially flat model, the rotor's turn angle is chosen to be the flat output and it is shown that all elements of the generator's state vector and the control input can be written as functions of the flat output and its derivatives. This procedure permits us to introduce a change of state variables (diffeomorphism) and to write the initial nonlinear model of the generator into the linear canonical form. For the linearized equivalent of the system, it is possible to design a state feedback controller with the use of pole-placement methods. Unlike linearization with the use of Lie derivatives, the flatness-based linearization approach does not require the computation of partial derivatives of the elements of the state vector or the computation of Jacobian matrices.

There are two particular issues in the design of the PMSG controller: (1) there are certain elements in the generator's state vector that are not directly measurable and (2) there may be variations in the mechanical input power (torque) and it may also be impractical or costly to measure this input power. To address point 1, a new nonlinear Kalman filtering approach is

introduced. The method is called derivative-free nonlinear Kalman filtering and consists of implementation of the standard Kalman filter recursion on the linearized equivalent of the PMSG model, which is obtained with the use of the differential flatness-based transformation. The fast computation features of the derivative-free nonlinear Kalman filter enable the estimation of non-measurable state vector elements in real time, while the accuracy of the provided state estimation is very satisfactory.

To address point 2, the derivative-free nonlinear Kalman filter is redesigned in the form of a disturbance estimator. The state vector of the disturbance estimator contains both the state vector elements of the linearized equivalent of the PMSG and new state vector elements that stand for the unknown mechanical input (torque) and its derivatives. Using the aforementioned disturbance observer, we can obtain estimates of both the non-measurable elements of the state vector (e.g. rotation speed and quadrature-axis transient voltage of the rotor) and estimates of the unknown disturbance input (which is due to external mechanical torque) and of its time derivatives.

To maintain the operating characteristics of the generator (e.g. rotation speed) unchanged despite variations in the input power, the initial state feedback controller of the PMSG is complemented by additional elements based on the disturbance input estimates that actually compensate for the unknown input effects. The considered disturbance estimator also makes it possible to compensate for the effects of parametric changes and modeling uncertainties in the PMSG dynamics. The efficiency of the proposed state estimation-based (sensorless control) of the PMSG is evaluated through simulation experiments.

12.10. Bibliography

- [ANE 07] ANENE E.C., ALIYA U.O., LÉVINE J., *et al.* “Flatness-based feedback linearization of a synchronous machine model with static excitation and fast turbine valving”, *IEEE 2007 Power Engineering, General Meeting*, Tampa, FL, June 2007.
- [BAS 93] BASSEVILLE M., NIKIFOROV I., *Detection of Abrupt Changes: Theory and Applications*, Prentice-Hall, 1993.
- [CHE 00] CHEN W.H., BALLANCE D.J., GAWTHROP P.J., *et al.*, “A nonlinear disturbance observer for robotic manipulators”, *IEEE Transactions on Industrial Electronics*, vol. 47, no. 4, pp. 932–938, 2000.
- [COR 05] CORTESAO R., PARK J., KHATIB O., “Real-time adaptive control for haptic telemanipulation with Kalman active observers”, *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 987–999, 2005.

- [COR 06] CORTESAO R., “On Kalman active observers”, *Journal of Intelligent and Robotic Systems*, vol. 48, no. 2, pp. 131–155, 2006.
- [DEL 01] DE LEON MORALERS J., BUSAWON K., ACHA-DUZA S., “A robust observer-based controller for synchronous generators”, *Electric Power and Energy Systems*, vol. 23, pp. 195–211, 2001.
- [FLI 99] FLIESS M., MOUNIER H., “Tracking control and π -freeness of infinite dimensional linear systems”, in PICCI G. and GILLIAM D.S., *Dynamical Systems, Control, Coding and Computer Vision*, vol. 258, Birkhäuser, pp. 41–68, 1999.
- [GUP 11] GUPTA A., MALLEY M.K.O., “Disturbance-observer-based force estimation for haptic feedback”, *ASME Journal of Dynamical Systems Measurement and Control*, vol. 133, no. 1, 2011, Article number 014505.
- [HAR 02] HARRIS C., HONG X., GAN Q., *Adaptive Modelling, Estimation and Fusion from Data*, Springer, 2002.
- [JIA 01] JIANG L., WU Q.H., ZHANG C., *et al.*, “Observer-based nonlinear control of synchronous generators with perturbation estimation”, *Electric Power and Energy Systems*, vol. 23, pp. 359–367, 2001.
- [KAM 99] KAMEN E.W., SU J.K., *Introduction to Optimal Estimation*, Springer, 1999.
- [KHA 96] KHALIL H.K., *Nonlinear Systems*, 2nd ed., Prentice Hall, 1996.
- [LAR 07] LAROCHE B., MARTIN P., PETIT N., “Commande par platitude: Equations différentielles ordinaires et aux dérivées partielles”, Ecole Nationale Supérieure des Techniques Avancées, Paris, 2007.
- [LEV 11] LÉVINE J., “On necessary and sufficient conditions for differential flatness”, *Applicable Algebra in Engineering, Communications and Computing*, vol. 22, no. 1, pp. 47–90, 2011.
- [LIN 10] LIN W.M., HONG C.M., CHENG F.S., “On-line designed hybrid controller with adaptive observer for variable-speed wind generation system”, *Energy*, vol. 35, pp. 3022–3030, 2010.
- [MAH 12] MAHMOUD M.A., ROTA H.K., HUSSAIN M.J., “Full-order nonlinear observer-based excitation controller design for interconnected power systems via exact linearization approach”, *Electric Power and Energy Systems*, vol. 41, pp. 54–62, 2012.
- [MAR 99] MARTIN PH., ROUCHON P., “Systèmes plats: planification et suivi des trajectoires”, *Journées X-UPS*, École des Mines de Paris, Centre Automatique et Systèmes, May, 1999.
- [MIK 06] MIKLOSOVIC R., RADKE A., GAO Z., “Discrete implementation and generalization of the extended state observer”, *Proceedings of the 2006 American Control Conference*, Minneapolis, MN, 2006.

- [NG 03] NG G.W., *Intelligent Systems - Fusion, Tracking and Control*, Research Studies Press, 2003.
- [RIG 01] RIGATOS G., ZHANG Q., Fuzzy Model Validation using the Local Statistical Approach, Publication Interne IRISA No 1417, Rennes, France, 2001.
- [RIG 07] RIGATOS G.G., TZAFESTAS S.G., "Extended Kalman filtering for fuzzy modelling and multi-sensor fusion", *Mathematical and Computer Modelling of Dynamical Systems*, vol. 13, pp. 251–266, 2007.
- [RIG 08] RIGATOS G.G., "Particle filtering for state estimation in industrial robotic systems", *IMEche Journal of Systems and Control Engineering*, vol. 222, no. 6, pp. 437–455, 2008.
- [RIG 09] RIGATOS G.G., "Particle filtering for state estimation in nonlinear industrial systems", *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 11, pp. 3885–3901, 2009.
- [RIG 10] RIGATOS G.G., "A derivative-free Kalman filtering approach for sensorless control of nonlinear systems", *IEEE International Symposium on Industrial Electronics IEEE ISIE 2010*, Bari, Italy, July 2010.
- [RIG 11a] RIGATOS G.G., "A derivative-free Kalman Filtering approach to state estimation-based control of nonlinear dynamical systems", *IEEE Transactions on Industrial Electronics*, vol. 59, no. 10, pp. 3987–3997, 2012.
- [RIG 11b] RIGATOS G.G., "Flatness-based adaptive fuzzy control for nonlinear dynamical systems", *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM) 2011*, Budapest, Hungary, July 2011.
- [RIG 11c] RIGATOS G.G., *Modelling and Control for Intelligent Industrial Systems: Adaptive Algorithms in Robotics and Industrial Engineering*, Springer, 2011.
- [RIG 11] RIGATOS G., AL-KHAZRAJI A., "Flatness-based adaptive fuzzy control for MIMO nonlinear dynamical systems", *Nonlinear Estimation and Applications to Industrial Systems Control*, Nova Publications, 2011.
- [RIG 12a] RIGATOS G.G., "Derivative-free nonlinear Kalman filtering for MIMO dynamical systems: application to multi-DOF robotic manipulators", *Journal of Advanced Robotic Systems* (special issue on Robot Manipulators), 2012.
- [RIG 12] RIGATOS G., SIANO P., ZERVOS N., "A distributed state estimation approach to condition monitoring of nonlinear electric power systems", *Asian Journal of Control*, 2012.
- [RUD 03] RUDOLPH J., "Flatness-based control of distributed parameter systems", *steuerungs- und Regelungstechnik*, 2003.
- [SIR 04] SIRA-RAMIREZ H., AGRAWAL S., *Differentially Flat Systems*, Marcel Dekker, New York, NY, 2004.

- [VIL 07] VILLAGRA J., D'ANDREA-NOVEL B., MOUNIER H., *et al.*,
 “Flatness-based vehicle steering control strategy with SDRE feedback gains
 tuned via a sensitivity approach”, *IEEE Transactions on Control Systems
 Technology*, vol. 15, pp. 554–565, 2007.
- [XIO 08] XIONG J., *An Introduction to Stochastic Filtering Theory*, Oxford
 University Press, 2008.

Chapter 13

Construction and Control of Parallel Robots

This chapter presents parallel manipulators, especially cable robots, as a new generation of cranes for which, all of six spatial degrees of freedom (DoFs) can be supported. The robot, its design and its manufacturing process are introduced from a mechatronics points of view. Actuating systems, required sensors, employed data transfer protocol and its designed graphical user interface (GUI) are discussed in this chapter for a parallel manipulator-like cable robot. The efficiency of the mentioned hardware setup and software installation, its advantages and drawbacks are focused and analyzed.

13.1. Introduction

Parallel robots, which are called kinematic machines, are a kind of closed-loop mechanism that have a high load carrying capability with an acceptable accuracy. Thus, a lot of applications can be mentioned for them in industrial environments like many kinds of simulators, cranes and automation lines. In a parallel mechanism, all of the actuators are connected to the end-effector and control it simultaneously. Moreover, for a parallel manipulator, there is a fixed platform by which the actuators are mounted and there is a moving platform, which is the end-effector of the robot, and the tool or load

is supposed to be installed on it and the manipulation is supposed to be performed through it.

There are different kinds of parallel robots depending on the type of actuators that are used by the robot to move the end-effector. The conventional type of parallel robots are made up of hydraulic actuators, and are called Stewart. The actuators can be also pneumatic or any combination of hydraulic, pneumatic and even electronic motors. A new generation of parallel robots that has been developed recently, is the cable robot in which the hydraulic jacks are replaced by cables. In these robots, the movement of the end-effector is provided by the aid of tensional force and the elongation of the cables, which are fulfilled using motors and pulleys. This idea was first proposed by National Institute of Standards and Technology (NIST) research center [ALB 93] and was developed rapidly in many academic and industrial centers. These robots have a good ratio of load to weight, high stiffness and perfect capability of assembling and de-assembling.

There are a number of robotic apparatus that use cables to position a platform. One of the most applicable types of underconstrained cable robots is cranes, which are widely used in industries. This idea is related to cable-suspended robots; the literature in this area started with the NIST RoboCrane [ALB 93]. Inspired by RoboCrane, many researchers have been involved with cable-suspended robots. A few of these have focused on cable-suspended crane devices. The typical cranes have the problem of stability due to their lack of actuating cables and cannot control the orientation of the suspended object. In the application of cranes, the designed cable robot is more trustworthy, though. This kind of underconstrained parallel mechanism, which is considered in this chapter, provides a practical mechanism for handling and positioning heavy loads while a larger useful workspace can be saved.

Aria *et al.* developed a seven-DoF, three-cable-suspended crane-type robot for an automobile assembly line, intended for heavy product assembly [OSU 93] and one mechanism with seven cables was also developed by Kawamura *et al.* [KAM 95]. Yamamoto *et al.* proposed a crane-type parallel mechanism with three active cables for handling heavy objects [YAM 99]. Shiang *et al.* presented a parallel four-cable positioning crane for offshore loading and unloading of cargo vessels under high seas [SHI 99]. Bostelman *et al.* changed the design from the symmetric RoboCrane structure to improve pose measurement and human interaction [BOS 95]. An experiment facility was developed at the University of Delaware that consists of a six-DoF cable-suspended robot with a reference governor [OH 05].

Another cable robot, which was designed to develop the cranes' applications, is the one that was manufactured at Pennsylvania University. This crane in contrast to the simple previous cranes has more cables to control the end-effector, which reduces the amount of end-effector's oscillations. There is no framework for this robot and all of the motors are mounted on the end-effector [CHA 04]. Two cable robots with an application in rehabilitation have been developed at Padua University. NeReBot is a three-DoF cable-based robot, designed for the treatment of patients with stroke-related paralysis or upper limb paretic during the acute phase [ROS 07]. Although the first clinical tests showed encouraging results in functional outcome, the robot presented some limitations. Therefore, a new cable-based robot was designed. The MariBot is a five-DoF serial-parallel robot. Like the NeReBot, three cables allow the movement of a patient's arm in the three-dimensional space by means of a splint. However, an additional two-DoF serial overhead structure is used to adapt a cable configuration to the current position of the patient's arm. This solution results in a significant increment of the workspace volume [ROS 06].

The kinematic calibration of a cable-actuated parallel robot is also discussed in [VAR 06]. Available redundant sensing of joint displacements makes it feasible to consider central linkage and cable actuators of the robot one at a time for calibration purposes. Dynamic characteristics of this type of mechanism such as the manipulation problem of load by multiple cables have been well studied [ALP 01, KOR 09].

In this chapter, formulation, design and manufacturing of these new type of parallel robots, that is cable suspended robots, are presented as a complicated sample of a mechatronic mechanism. First, the overall scheme of its mechanical structure is presented. Afterwards, its formulation including kinematics, kinetics and control is derived. Then, its corresponding hardware setup and installation including actuators, sensors and data transfer protocol is explained. The required electrical boards and drivers and their corresponding calibrations are presented. Finally, its related GUI and simulator, which are programmed in LabView, are described as the required software implementation.

13.2. A parallel robot mechanism

An underconstrained cable robot is designed and manufactured at Iran's University of Science and Technology (ICaSbot), which supports six DoFs including three translational and three rotational movements of the

end-effector with the aid of six active cables and six DC motors. A setup of the above-mentioned robot can be seen in Figure 13.1 [KOR 11a]. The experimental tests are conducted on this robot in order to verify the efficiency of the proposed force measurement sensors and used them in the mentioned applications. A hardware setup and supporting software are designed and programmed for evaluating the required parameters of load cells.



Figure 13.1. *The ICaSbot robot*

Six DC motors are responsible for supporting six suspended cables. The change in length of these cables controls three translational and three rotational movements of a triangular shape end-effector. The cable, which transfers the weight load, is passed over the pulley and is wrapped around the shaft that is controlled by the aid of motors. Other specifications of the robot are listed in Table 13.1. The assembled system of encoder, shaft and motor is shown in Figure 13.2 [KOR 11a].

13.2.1. Mechanical structure

The main part of the robot mechanical features consists of an end-effector supported by six actuating cables, which can be retracted and extracted through their related pulleys. These pulleys are also connected to six drums and corresponding motors. The encoders are attached to the couples of the drums and motors. They are responsible for counting the rotation of each motor. Figure 13.1 shows the scheme of the Iran University of Science and Technology (IUST) built cable robot and Table 13.1 shows its main geometrical specifications. One end of the cables is attached to the pulleys

whereby the platform is suspended in the workspace. The other end is anchored to the end-effector and the cables are made taut by the actuator system. Each of them is adapted for generating a position signal that is responsible for retraction and extension of the cables, and then the central control system is able to provide commands for translating and orienting the platform with the aid of combining these individual signals related to each cable [KOR 11a].

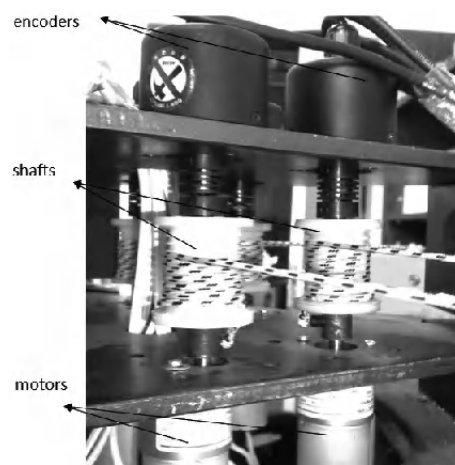


Figure 13.2. *The assembled system, from top to bottom: (1)encoder, (2) shaft, (3) motor*

Robots Specifications	Value
Height	120 cm
Variable side length of base triangle	100–120 cm
Weight	350 kg
End-effector	
Side length of base	17 cm
Thickness	8 mm
Weight	1,100 g

Table 13.1. *The specifications of the robot [KOR 11a]*

This section also presents a system of Cartesian pose measurements using a camera installed on the robot structure. The proposed system is relatively simple and economical [KOR 11a].

The actuator system is located on the base platform and includes three separate important parts, that is: motors, encoders and drums. Therefore, the cables can be retracted or extended by this actuator system in order to suspend, translate and orient the platform as desired [KOR 11a].

On the other hand, the control system is designed so that the platform could be translated and rotated in the workspace by controlling the length of the cables extending from their respective pulleys. A designed electronic board controls motion of each cable by calculating and imposing its desired length. Using at least six cables, the platform has six DoFs: translation in three axes and rotation about each of the foregoing three axes. Numerous types of end-effectors can be mounted on the platform, thus enabling the robot to perform a wide variety of tasks. The robot as described here does not require a separate rotational device on the platform to orient the end-effector in the workspace. Figure 13.3 illustrates the sequential procedure of power transmission of the manufactured robot and Figure 13.4 is a preview of the used end-effector with its lasers connected to it, which illustrates the position of the moving platform with the aid of a vision system [KOR 11a].

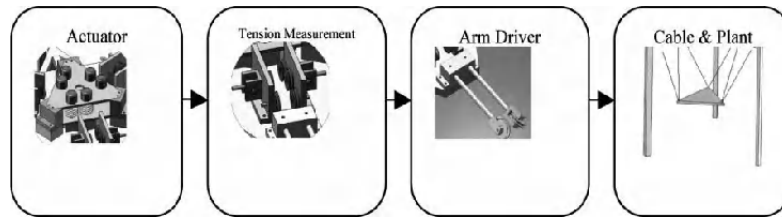


Figure 13.3. Sequential procedure of power transmission [KOR 11a]



Figure 13.4. Used end-effector

13.2.2. Modeling and Formulation

13.2.2.1. Kinematics

The overall scheme of a ICaSbot cable-suspended robot is shown in Figure 13.5 [ALP 01].

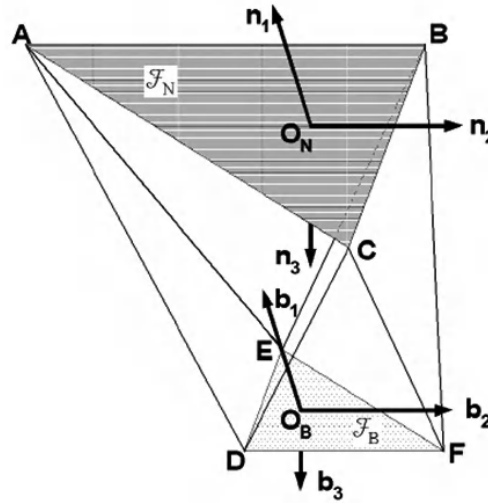


Figure 13.5. Schematic of six cable robots [ALP 01]

where $x_m, y_m, z_m(m)$ are translational coordinates of the end-effector and $\psi, \theta, \phi(\text{rad})$ are its rotational coordinates. Presenting the cables' elongation with L , the following relation can be established between the elongation of the cables and end-effector's velocity [IRA 09]:

$$\dot{L} = J \begin{bmatrix} \dot{X} \\ \omega \end{bmatrix} = J \begin{bmatrix} \dot{x}_m, \dot{y}_m, \dot{z}_m, \dot{\psi}, \dot{\theta}, \dot{\phi} \end{bmatrix}^T \quad [13.1]$$

$$J = \begin{bmatrix} \frac{\partial L_i}{\partial X} \end{bmatrix} \quad [13.2]$$

where J is the Jacobian matrix. On the other hand, the following equation can be expressed between the cables' elongation L and pulleys' angular rotation β :

$$\beta = \begin{Bmatrix} \beta_1(X) \\ \vdots \\ \beta_n(X) \end{Bmatrix} = \frac{1}{r} \begin{Bmatrix} L_{01} - L_1(X) \\ \vdots \\ L_{0n} - L_n(X) \end{Bmatrix} \quad [13.3]$$

where r is the pulley radius. As is shown in Figure 13.6, calculating pulleys' angular velocity from a path is called inverse kinematics and obtaining the path from a pulleys' angular velocity is called direct kinematics [KOR 11d].

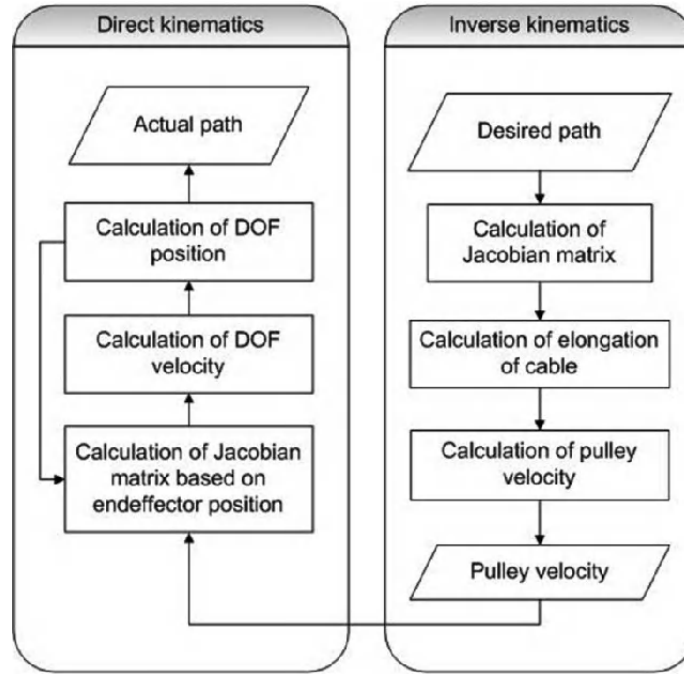


Figure 13.6. Kinematics chart [KOR 11d]

13.2.2.2. Dynamics

The use of a Newton–Euler equation leads the system to have the following dynamic equation [IRA 09]:

$$D(X)\ddot{X} + C(X, \dot{X})\dot{X} + g(X) = -J^T(X)T \quad [13.4]$$

where:

$$D = \begin{bmatrix} mI_3 & 0 \\ 0 & P^T IP \end{bmatrix}; C = \begin{bmatrix} 0_3 \\ P^T \{I\dot{P}\dot{o} + (P\dot{o}) \times I(Po)\} \end{bmatrix}; g = \begin{bmatrix} 0 \\ 0 \\ -mg \\ 0_3 \end{bmatrix}; \quad [13.5]$$

$$J = \left[\frac{\partial q_i}{\partial x_j} \right]_{i \times j}; P = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \psi & \sin \psi \cos \theta \\ 0 & -\sin \psi & \cos \psi \cos \theta \end{bmatrix}; \dot{o} = \begin{bmatrix} \dot{\Psi} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix}$$

T is the vector of the cables tension, D(x) is the inertia matrix, C(x, $\dot{x}$) is the vector of velocity terms, g(x) is the gravity vector, J is the conventional parallel manipulator Jacobian, X is the vector of DoFs of the system, m is the mass of the end-effector, I is the moment of inertia of the end-effector and q is the length of the cables. Also, the dynamics of the motor is as follows:

$$T = \frac{1}{r} \left[\tau - j \left(\frac{d}{dt} \left(\frac{\partial \beta}{\partial X} \right) \dot{X} + \frac{\partial \beta}{\partial X} \ddot{X} \right) - c \frac{\partial \beta}{\partial X} \dot{X} \right] \quad [13.6]$$

where j is the matrix of rotary inertia of the motors, c is the viscous friction matrix of the motors, $\dot{\beta}$ is the vector of angular velocity of the motors and τ is the vector of motors torque. By coupling these two dynamics, we have:

$$D(X) \ddot{X} + C(X, \dot{X}) \dot{X} + g(X) = -J^T(X) \frac{1}{r} \left[\tau - j \left(\frac{d}{dt} \left(\frac{\partial \beta}{\partial X} \right) \dot{X} + \frac{\partial \beta}{\partial X} \ddot{X} \right) - c \frac{\partial \beta}{\partial X} \dot{X} \right] \quad [13.7]$$

As is shown in Figure 13.7, calculating motor's torque from the given path is called inverse dynamics and obtaining the actual path from the applied torque is called direct dynamics.

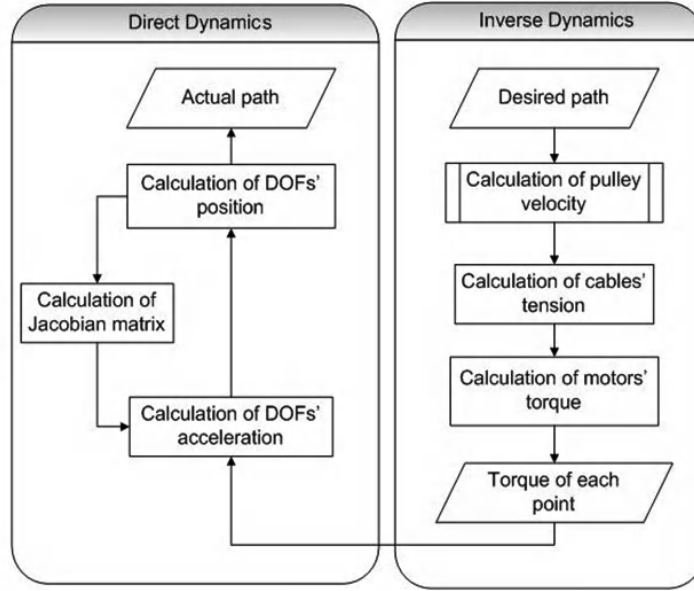


Figure 13.7. Dynamics chart [KOR 11d]

13.2.2.3. Control Scheme

[KOR 12c] using the feedback linearization method results in the following required cables' tension:

$$T_i = \{ J^{-1}(D(X)v + C(X, \dot{X}) + g(X)) \}_i; \quad i = 1, \dots, 6 \quad [13.8]$$

V is the control input of the feedback linearization and can be substituted as below to create a controllable error equation:

$$v_i = \ddot{X}_d + K_{iD}(\dot{X}_d - \dot{X}_a) + K_{iP}(X_d - X_a); \quad i = 1, \dots, 6 \quad [13.9]$$

where K_{iD} and K_{iP} are controlling gains of derivative and proportional errors, respectively, X_d is the desired trajectory of the robot and X_a is its actual value.

So, the required cable tension is calculated here using inverse kinetics of the robot together with the feedback linearization method and the required

angular velocity of the motor is evaluated using inverse kinematics of the robot. It is now possible to study the dynamics of the motor in order to evaluate its required Pulse Width Modulation (PWM). Therefore, the dynamic of the motor can be stated as [KOR 12a]:

$$\tau_i = rT + J\ddot{\beta} + c\dot{\beta}; \quad i = 1, \dots, 6 \quad [13.10]$$

Here, the desired motor torque and angular velocity are calculated as in the previous section using the inverse dynamic of the motor, which results in the required torque of the motors (computed torque method + feedback linearization). So, substituting equation [13.8] in equation [13.10] results in:

$$\tau_i = rJ^{-1}(Dv + C + g) + J\ddot{\beta} + c\dot{\beta}; \quad i = 1, \dots, 6 \quad [13.11]$$

Generally, the overall scheme of controlling the robot in a closed-loop way is shown in Figure 13.7. It can be seen that two sequential closed-loop controllers are involved: motor controller and end-effector controller. The angular velocity of the motors and the torque of the motor are controlled using encoder and load cell sensors for each 0.01 s in the inner loop of robot control while the position of the end-effector is controlled for each 0.1 s using the position feedback of lasers and camera sensors.

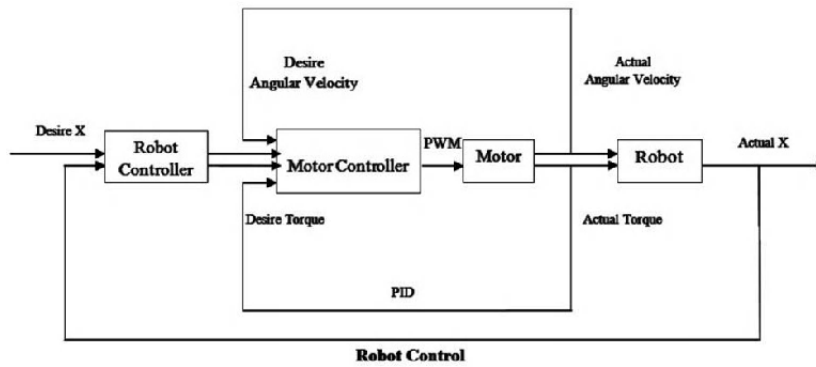


Figure 13.8. Diagram of the proposed control system for ICaSbot

13.3. Actuators

13.3.1. Motor

Six motors were selected for the robot whose specifications are listed in Table 13.2. Each motor is connected to an encoder, which has a high resolution precision, by a 3 cm-diameter circular shaft. The cable that transfers the weight load is passed over the pulley and wrapped around the shaft. As Table 13.2 shows, this 12 V 17 W geared motor runs at 150 rpm (in free running condition).

Motor's specifications	Value
Mark	Retarding gear motor
Model	1.61.070.304 Buhler
Nominal voltage	12 V
No load speed	150 rpm
Stall torque	1.7 N·m
No load current	0.3 A
Stall current	1.2 A
Reduction ratio	1:175
Weight	220 g

Table 13.2. *DC motor's specifications*

The dynamic of the motor can be stated [13.11], where the applied torque of the motor is τ , J is the rotary inertia of the motor, r is the radius of the drum, $\dot{\beta}$ is its angular velocity and c is the viscose damping of the motor. Here, the desired motor torque and angular velocity are calculated as in the previous section using the inverse dynamic of the motor, which results in the required torque of the motors (computed torque method + feedback linearization).

Here, DC motors are used to provide the desired calculated torque and angular velocity of the motors. The goal of DC motor modeling is to extract a proper formula presenting a relation between the armature voltage and the relative produced torque. Figure 13.10 shows the basic circuit diagram of a DC motor.

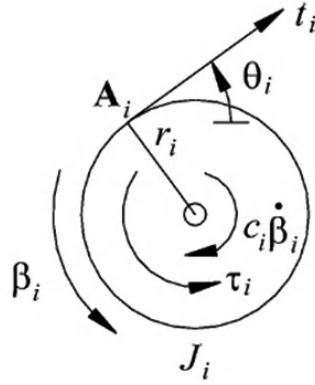


Figure 13.9. Scheme of the motor [ALP 01]

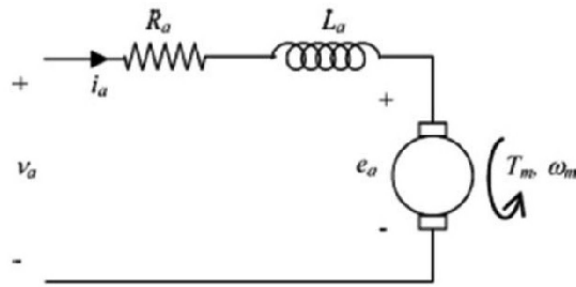


Figure 13.10. Circuit diagram of a DC motor

The general equations of Direct Current Permanent Magnet (DCPM) motor are:

$$v_a = R_a i_a + L_a \left(\frac{di_a}{dt} \right) + e_a \quad [13.12]$$

$$e_a = K_m \omega_m \quad [13.13]$$

$$\tau_m = K_m i_a \quad [13.14]$$

where v_a is armature voltage, R_a is the resistance of armature wire, L_a is armature inductance, i_a is armature current, e_a is emf (induced voltage of

electromotor force), K_m is constant of torque motor, τ_m is produced torque and ω_m is angular velocity of the DC motor [KOR 12a]. So the required voltage of the motor can be computed based on desired torque and speed of the motor using equation [13.12], where e_a is substituted by equation [13.13] and i_a is substituted by equation [13.14]. Finally, this voltage can be implemented on the motor using a PWM method that has the least loss of energy.

$$PWM = v_a / v_{\max} \quad [13.15]$$

where $v_{\max}$ is the maximum voltage of the motor. However, this is possible if and only if the exact model and parameters of the motor are available and it is obvious that in many conditions these parameters are not available. Finally, PWM is the required pulse width modulation of the DC motor.

The required calculated torque of the motors is provided here by the aid of the PWM method. The feed-forward term of PWM is calculated using the inverse dynamics while the errors are compensated using a PD controller on the velocity of the motors:

$$\begin{aligned} PWM = & PWM_f + K_d \times (\dot{\beta}_d - \dot{\beta}_a) + K_p \times (\beta_d - \beta_a) \\ & + K'_d \times (\dot{\tau}_d - \dot{\tau}_a) + K'_p \times (\tau_d - \tau_a) \end{aligned} \quad [13.16]$$

where d indicates the desired output and a is the actual output. The coefficients K_d and K_p are controlling gains of the proportional and the derivative error of the motor angular velocity and K'_d and K'_p are the controlling gains of motor torque. Our primary goal is to show the actual amount of torque and tension of the cables to be used as a feedback in controlling the torque of the motors in order to increase the accuracy. Therefore, not only can the PWM be improved using the encoder feedback of kinematic response of the motor speed (which helps the system to achieve the desired speed), but also its desired torque can be provided using the feedback of the actual motor torque.

In this test, the PWM method is used for regulating the input voltage. In the PWM method, the desirable pulse width is amplified by a motor driver IC, L298. The electronic circuit amplifies the PWM signal generated from the data acquisition board. The lower bound of the PWM is related to 0 V and the

upper bound is related to 12 V. In general, the larger command signal produces the wider pulse. The motor-driven circuit is shown in Figure 13.11.

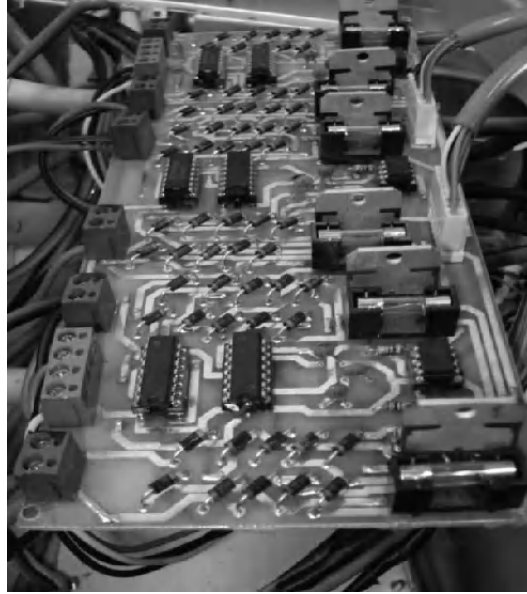


Figure 13.11. *Motor driver circuit*

13.3.2. Cables

Moreover, the used cables are not completely solid and can be easily vibrated in both longitudinal and lateral dimensions. It is obvious that these kinds of vibrations can enormously affect the desired position and rotation of the end-effector. Configuration of cables and their connection points to the end-effector are shown in Figure 13.4 while their specifications are explained in Table 13.3.

Cable's specification	Value
Diameter	3 mm
Elasticity module	$5.6 \times 10^6 \text{ N/m}^2$

Table 13.3. *Cable specifications*

During a robot operation for carrying a payload in a workspace, the end-effector position measurement can be considered in some ways. Using encoders for the motors on the active cables can result in some errors related to cable sag and stretch. The ICaSbot design allows the moving cable to rotate more than one time around a same line of the drum, resulting in further measurement error. The motors actuate the active driven cables and the control system commands the motors.

13.4. Sensors

In order to control the robot, there are three different kinds of sensors, which include encoders, load cells and also the combination of a camera and optical sensors.

13.4.1. *Motor speed sensor*

The incremental encoders used in this hardware setup have resolution precision, 4×600 pulses per revolution; two output signals of encoders have 90° phase shift. The encoder's specifications are shown in Table 13.1.

Encoder's specifications	Value
Mark	Autonics
Model	E50S8-600-3-T-24
Resolution	600
Output phase	A, B, Z
Control output	Totem pole
Power supply	12–24 V
Maximum allowable revolution	5,000 rpm

Table 13.4. *The specifications of the selected encoder*

There is a possibility of connecting the encoders to the data acquisition board directly without using the designed intermediate printed circuit board (PCB) and estimating the angular velocity and the motion direction using the phase shift between two output signals; however, we prefer to use the

designed intermediate PCB in order to increase the precision and reduce the amount of mathematical calculation.

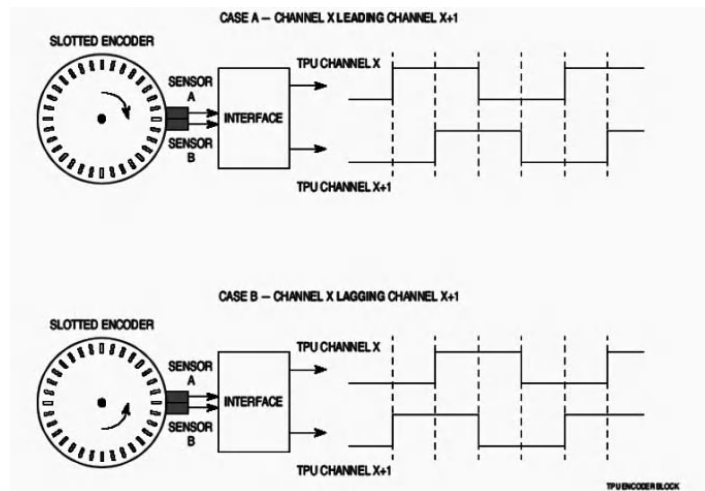


Figure 13.12. Output of encoder pulse

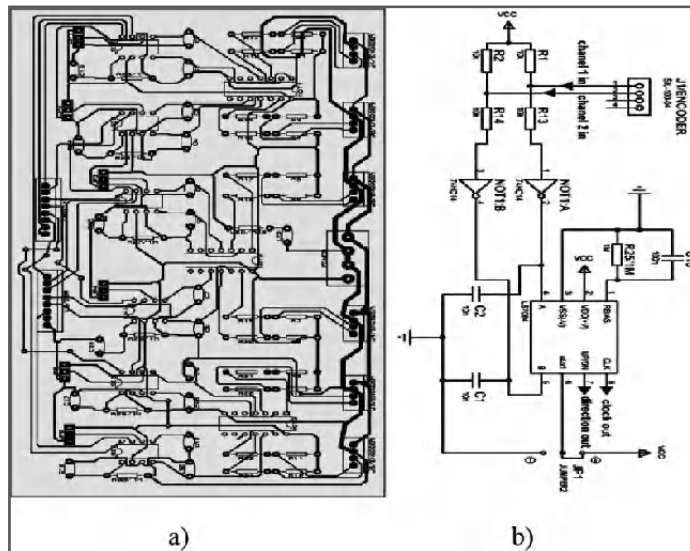


Figure 13.13. a) Circuit diagram of the PCB reads rotary encoder's output and b) its layout

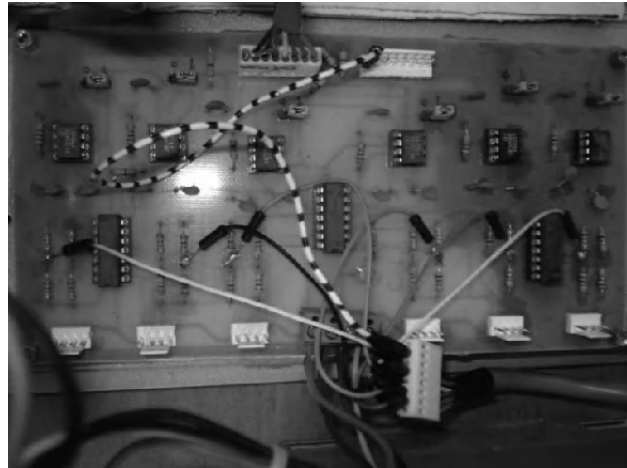


Figure 13.14. *Rotary encoder interface circuit*

The designed PCB increases the reading precision from 600 to 2,400 pulses per revolution and the output signal received by the data acquisition board is a pulse that varies between 0 and 1 V. By using the software setup, the received signal can be read. The electronic circuit read rotary encoder's output and its layout is shown in Figure 13.13. The PCBs, data acquisition board's ports and the power supply are installed in a hardware setup box. Figure 13.13 shows a schematic circuit and PCB read rotary encoder's output.

13.4.2. Force sensor

The torque measurement mechanism is designed in such a way that the force applied to the pulley is transferred to a load cell by a shaft. Cable tension feedback can be evaluated accurately for precise movements. As is shown in Figure 13.15, the mechanism consists of three pulleys that are installed on the structure by means of three shafts. The two lateral pulleys are fixed and they can only rotate to transfer the cables, whereas the central pulley can not only rotate but can also move vertically up to 5 cm. This pulley is installed on the force sensor by bolts. When two lateral pulleys transit the cables, the central pulley can move. The shaft connected to the force sensor can transfer the applied force to the sensor. The more the force, the greater the pulley movement and the greater the force sensor output.

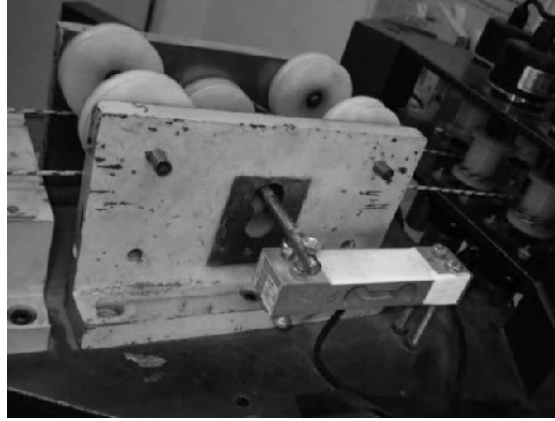


Figure 13.15. *Load cell as the used tension sensing device*

The used force sensor (load cell) type is a single-point aluminum alloy. As is shown in Figures 13.15 and 13.16, one side is connected to a pulley's shaft while the other side is connected to the end-effector by means of a bolt. This sensor consists of four strain gauges that convert the vertically applied force to voltage. Each resistance bridge (Figure 13.17), including strain gauges, magnifies the voltage up to 5–10 V DC. The capacity of each sensor is up to 5 kg and the accuracy is $0.1 \text{ V} + 0.2 \text{ mV/V}$ [TAJ 11]. Specification of each load cell is listed in Table 13.2.



Figure 13.16. *Force sensor that is connected to torque measurement mechanism*

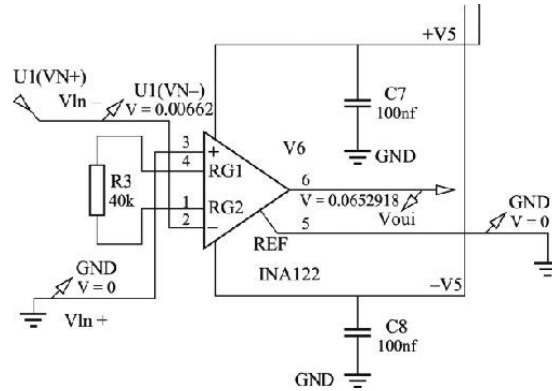


Figure 13.17. The schematic circuit of the force sensor's output amplifier

Model	Single Point 1661
Material	Aluminum Alloy
Surface	anodized treatment
Size	$250 \times 350 \text{ mm}^2$
Capacity	5 kg
Excitation Voltage	5–10 V dc
Output sensitivity	$2.0 \pm 0.1 \text{ mv/v}$

Table 13.5. Specifications of the force sensor

An “Advantech-PCI-1780” board is applied for reading the data of the motor and the encoder sensor (Figure 13.8). This board has eight channels and uses an external electronic board (PCLD-8710-AE) and it has specific cables (PCI-10168) to reduce the noises.

A proper data transfer system is required to manage the data transfer rate. Hence, an A/D sampling card is used. The output voltage of the load cell is approximately 0.2 mV/V. In order to measure this voltage, noise reduction and voltage amplification are necessary. To overcome this problem, an amplifier and noise reducer electronic board is constructed. This board consists of an amplifier IC (INA114) possessing a negative feedback to reduce noises. Besides, the negative feedback INA114 has some advantages including a high signal-to-noise ratio (SNR), high and variable gain and low

cost. The simplified schematic circuit of the electronic board and its PCB is illustrated in Figure 13.18. Here, U are the amplifier ICs of INA114 and the reducer of the noises, C s are installed capacitors, $J3$ is the input voltage of the board, $J1$ and $J2$ are output channels to the data card and, eventually, $R1$ – $R6$ are the employed regulator resistance for the used ICs.

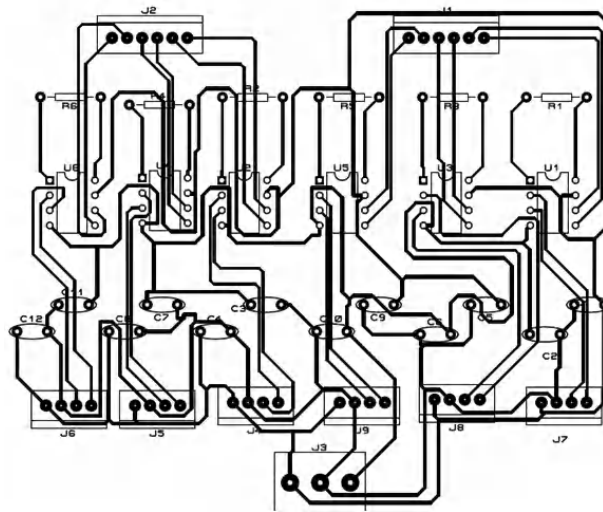


Figure 13.18. The PCB of force sensor's output amplifier circuit

Equations [13.17] and [13.18] calculate the INA114's output voltage and gain, respectively. The gain value depends on the resistance gain (R_G) and it should be regulated from the excitation table in the INA114 datasheet. V_O is the force sensor output voltage; V_{IN}^+ and V_{IN}^- are force sensor excitation voltage and G is the force sensor amplification value that varies with variation in resistance on board.

$$V_O = G \times (V_{IN}^+ - V_{IN}^-) \quad [13.17]$$

$$G = 1 + \frac{50 \text{ k}\Omega}{R_G} \quad [13.18]$$

The final PCB board used for recording the load cell's output is shown in Figure 13.19.



Figure 13.19. *The final PCB board for recording the loadcell output*

To control and record data, a 100 kHz analog sampling card (Advantech PCI 1711L) is used, which transfers the force sensor output to a PC. This card is an interface between software and amplifier board [TAJ 11].

13.4.3. *Position sensor*

The third type of sensors is the combination of a camera and optical sensors.

A camera is installed at the top of the end-effector, which is able to record the virtual (raw) data of the planar coordinate of the end-effector. The specifications of the used camera are listed in Table 13.6.

Camera parameters	Value
Category	Webcam
Video capture (4:3)	640×480
Image capture	640×480
Diagonal field of view (FOV)	68°
Connection type	USB 2.0
Width	40.4 mm (1.59 in)
Depth/length	68.5 mm (2.69 in)
Weight	88 g (3.12 oz)

Table 13.6. *Specification of the used camera*



Figure 13.20. *a) Optical sensors. b) Camera*

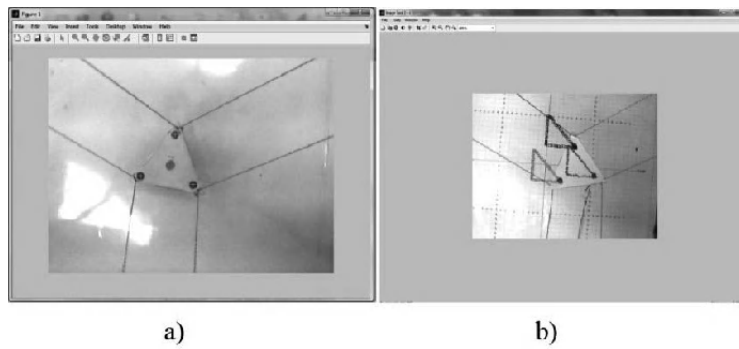


Figure 13.21. *a) Image preprocessing and b) image processing*

Two series of calibration are required for the recorded data of the camera to achieve the real planar coordinate of the end-effector. The first one is a planar calibration of the camera in which the origin of the coordinate system is determined for a specific altitude of the end-effector. The second one is perspective calibration that needs the instantaneous altitude and orientation of the end-effector. These data are fed by the aid of installed lasers that have been discussed earlier. A chess-shaped plate is used to perform the planar calibration of the camera. Two predefined points are specified in this plate for which their real planar coordinates are known. By running the program, the positions of three markers are first selected manually by clicking on the screenplay of the software. Then, the software starts recording the position of the three markers that have been selected.

Next, the program converts the raw data achieved in image processing to real data by using a perspective equation of the installed camera. For

converting raw data, the height of the markers are required, which are obtained by three laser sensors installed on the bottom face of the end-effector.

$$L_{\text{real}} = \frac{L_{\text{virtual}} \times (h + f)}{f} \quad [13.19]$$

Figure 13.22 is used for converting the virtual or raw coordinate data to real data. Here, h is the distance between the camera and the point, and f is canonical distance of camera, which can be obtained by an experiment as discussed below.

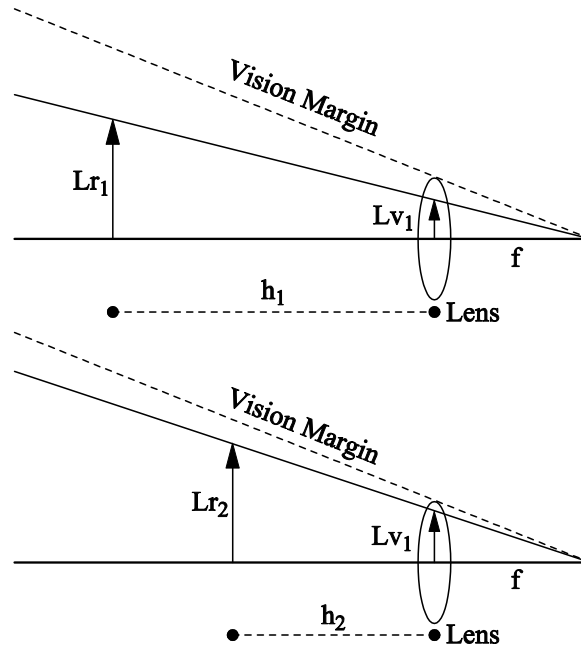


Figure 13.22. The scheme of the reflection of the object on the lens of the camera (perspective effect)

For obtaining the f value of the camera, a specific object should be placed at two predefined distances h_1 and h_2 from the camera lens. Using [13.20], the physical parameter of the camera, f can be obtained. By substituting the obtained f in [13.20], the perspective equation is calibrated.

$$\begin{cases} \frac{L_{r1}}{f} = \frac{L_{r1}}{h_1 + f} \\ \frac{L_{r2}}{f} = \frac{L_{r2}}{h_2 + f} \end{cases} \quad [13.20]$$

Three laser sensors of type GP2D12 are installed on the each vertex of the end-effector triangle. The specifications of the used optical laser sensor are listed in the Table 13.7.

Optical laser sensor parameters	Value
Output	Analog
Effective range	10–80 cm
LED pulse cycle duration	32 ms
Typical response time	39 ms
Typical start up delay	44 ms
Average current consumption	33 mA
Detection area diameter at 80 cm	6 cm
Supply voltage	–0.3 to +7.0 V
Output terminal voltage	–0.3 to Voltage at the Common Collector (VCC + 0.3) V
Operating temperature	–10 to +60°C
Storage temperature	–40 to +70°C

Table 13.7. Specification of the used optical laser sensor

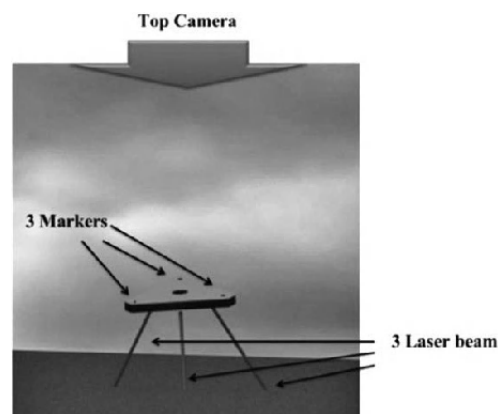


Figure 13.23. Symbolic scheme of laser beam

This sensor has one light emitter and one light receiver, which are placed on the base. The light emitter unit emits a light beam with specific frequency toward the object and receives that specific light reflection by light receiver unit. Three laser sensors are placed under the end-effector aiming at the ground.

Sensors are calibrated manually, since it was observed that the sensors do not confirm the calibration function of their catalog (Figures 13.24 and 13.25).

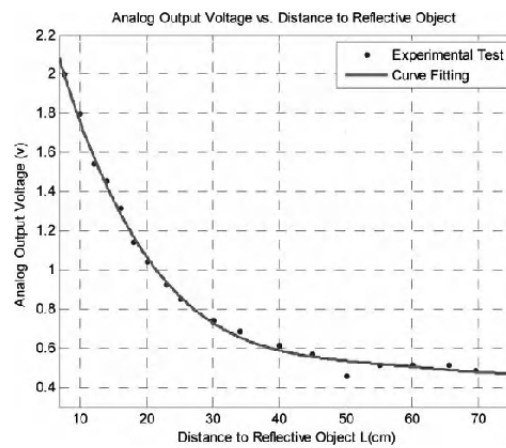


Figure 13.24. Calibration test related to laser sensor experiment data

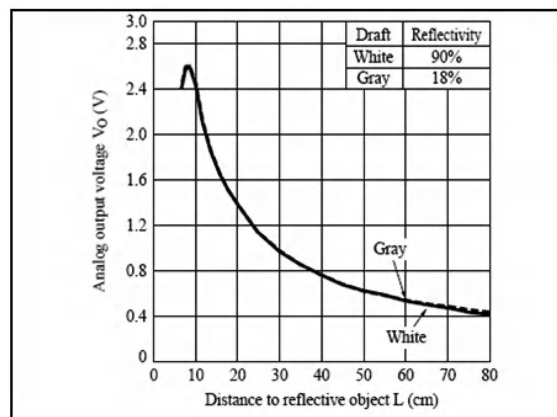


Figure 13.25. Calibration test related to laser sensor catalog data

In this step, three lasers are installed at three tips of the end-effector with a specific angle by which their laser beams eventually configure a half pyramid. It is required first to evaluate the translational position of the vertexes of the end-effector, and then the rotational movement can be determined in an online way. Since the length of the edges of the pyramid can be determined by the aid of lasers data, the geometry of the pyramid will be known completely by which the altitude of three tips can be calculated easily by the aid of following formula.

$$\begin{aligned}
 X_{\alpha} &= \sin(\zeta)L_{\alpha}\cos(30) + d/2 \\
 Y_{\alpha} &= -\sin(\zeta)L_{\alpha}\sin(30) - d/(2\sqrt{3}) \\
 Z_{\alpha} &= -\cos(\zeta)L_{\alpha} \\
 X_{\beta} &= -\sin(\zeta)L_{\beta}\cos(30) - d/2 \\
 Y_{\beta} &= -\sin(\zeta)L_{\beta}\sin(30) - d/(2\sqrt{3}) \\
 Z_{\beta} &= -\cos(\zeta)L_{\beta} \\
 X_{\delta} &= 0 \\
 Y_{\delta} &= \sin(\zeta)L_{\delta} + d/(\sqrt{3}) \\
 Z_{\delta} &= -\cos(\zeta)L_{\delta}
 \end{aligned} \tag{13.21}$$

where X, Y, Z are in local coordinate system, d is the length of the end-effector triangle edge (Figure 13.27) and ζ , which is the angle between the direction of the laser beams and local Z axis, is equal to 30° for the mentioned installation of the ICaSbot. Equation [13.11] is presented in a local coordinate system. By calculating the local coordinate of three mentioned intersection points, the equation of the plane (Figure 13.28), which includes these three points, can be obtained:

$$a(X - X_0) + b(Y - Y_0) + c(Z - Z_0) = 0 \tag{13.22}$$

where a, b and c are the resultant vectors of the cross products of $\overrightarrow{\alpha\beta} \times \overrightarrow{\alpha\delta}$ and X_0, Y_0, Z_0 are the local coordinates of one of the tips (e.g. α point). Now by substituting the local X, Y of each marker point (R, L and U points) in this plane formula, the vertical distance of each tip with respect to the end-effector angle can be easily calculated, which is called Z . Finally, the vertical altitude of each tip from the ground (h) can be determined by multiplying this distance by $\cos \gamma$.

$$h = z \cos(\gamma) \quad \cos \gamma = \frac{c}{\sqrt{a^2 + b^2 + c^2}} \quad [13.23]$$

where γ is the angle of the normal vector of the end-effector compared to vertical axis. Now the scaling of the image processing can be completed by having the altitude of each tip of the end-effector. The real coordinate of each tip can be obtained by the aid of the raw data of the camera together with the resultant altitude of each tip. Finally, the coordinate of the end-effector centroid can be evaluated by the aid of the real calculated coordinate of three tips of the end-effector:

$$\hat{X}_G = \frac{\hat{X}_L + \hat{X}_R + \hat{X}_U}{3} \quad \hat{Y}_G = \frac{\hat{Y}_L + \hat{Y}_R + \hat{Y}_U}{3} \quad \hat{Z}_G = \frac{\hat{Z}_L + \hat{Z}_R + \hat{Z}_U}{3} \quad [13.24]$$

where $\hat{X}, \hat{Y}, \hat{Z}$ are in the global coordinate system and the index G denotes the centroid coordinate of the end-effector. Alternatively, the global angles of the end-effector can be calculated using [13.24] and the following formulations:

$$\begin{aligned} A &= \frac{a}{\sqrt{a^2 + b^2 + c^2}} \\ B &= \frac{b}{\sqrt{a^2 + b^2 + c^2}} \\ C &= \frac{c}{\sqrt{a^2 + b^2 + c^2}} \end{aligned} \quad [13.25]$$

Considering a 30° angle for each laser with respect to vertical axis (Figure 13.27), the length of edges $L_\alpha; L_\beta; L_\delta$ is obtained (Figure 13.26). A local coordinate is set in the centroid of the end-effector and the local X, Y, Z of the points of intersection of the lasers with the ground can be calculated by measuring the angle of the lasers with respect to the vertical axis.

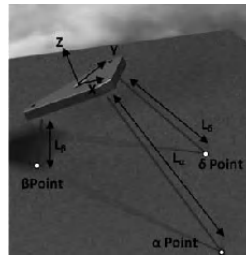


Figure 13.26. Length of laser beams and their related intersection points with ground

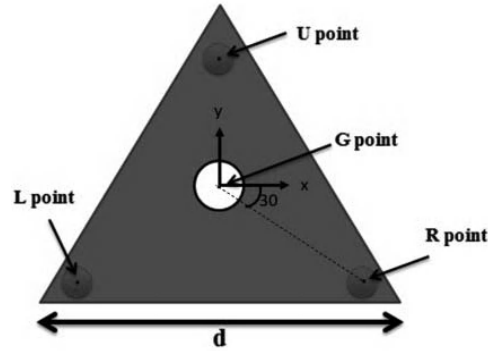


Figure 13.27. Markers' points and center point of the end-effector

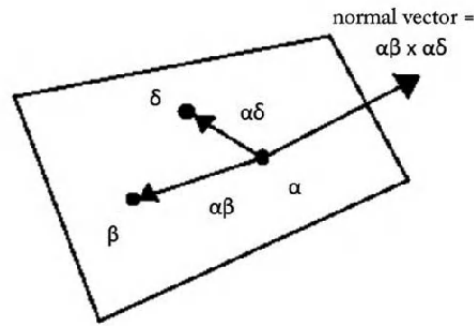


Figure 13.28. The equation of a plane passing through three points

Now, as the exact translational position of the three vertexes of the end-effector is determined, it is possible to calculate the instantaneous rotational condition of the end-effector, which is done in an online way. The angles of the end-effector can be calculated as:

$$\text{Yaw} = \tan^{-1} \left(\frac{A}{-B} \right) \quad [13.26]$$

$$\text{Pitch} = \frac{\sqrt{A^2 + B^2}}{C} \quad [13.27]$$

Moreover, the roll can be obtained by the aid of vision data as follows:

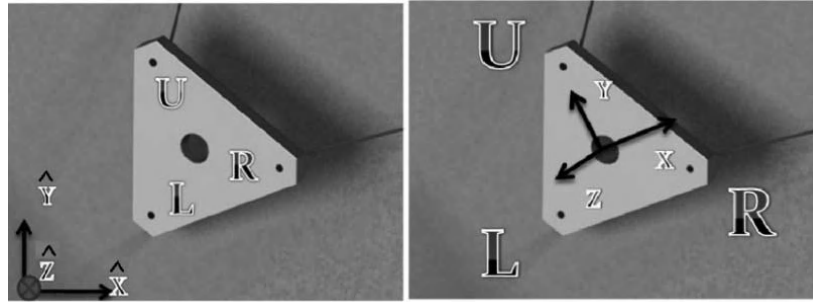


Figure 13.29. *Global coordinate and local coordinate*

Since the global $\hat{X}$, $\hat{Y}$ coordinates of three markers are known by the use of the camera (the global coordination of U-marker: $(\hat{X}_U, \hat{Y}_U)$ and also for R-marker and L-marker: $(\hat{X}_R, \hat{Y}_R)$ and $(\hat{X}_L, \hat{Y}_L)$), we can evaluate the roll by the aid of following equation:

$$\text{Roll} = \tan^{-1} \left(\frac{\hat{Y}_R - \hat{Y}_L}{\hat{X}_R - \hat{X}_L} \right) \quad [13.28]$$

While the robot is moving, the data of $\hat{X}$, $\hat{Y}$ and $\hat{Z}$ position and also three axial orientations are recorded in a matrix in Matlab[®]'s workspace to be used as feedback control for each time step. The whole explained formulation can also be developed for the environmental situations in which the ground is not flat. This purpose is easily possible by either replacing the ground with a new flat surface or extracting the predefined surface formulation of the ground. Also, this limitation that the end-effector needs to be always parallel to the ground is canceled here since the procedure is done for all of the three vertexes of the end-effector and is analyzed in the mentioned formulations.

13.5. Data transfer protocol

Also, two kinds of data card of Advantech Company are used to control the robot.



Figure 13.30. *PCI 1711 Advantech data acquisition board*

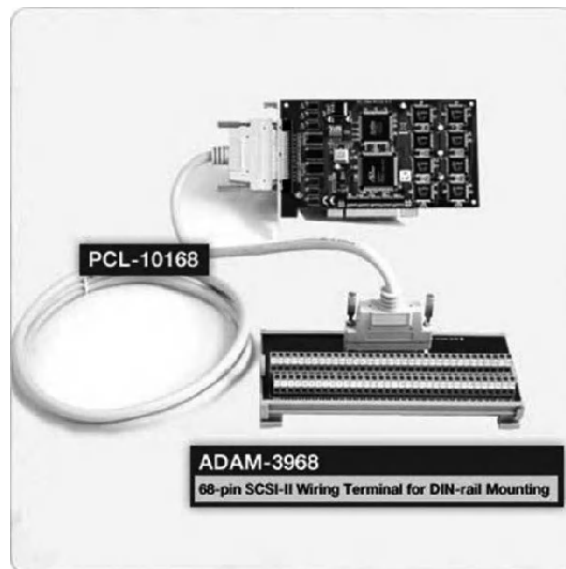


Figure 13.31. *PCI 1780 Advantech data acquisition board*

The GUI software setup that reads the encoders' output signals using the data acquisition board is programmed in LabVIEW environment. A panel of the GUI software setup is shown in Figure 13.32.

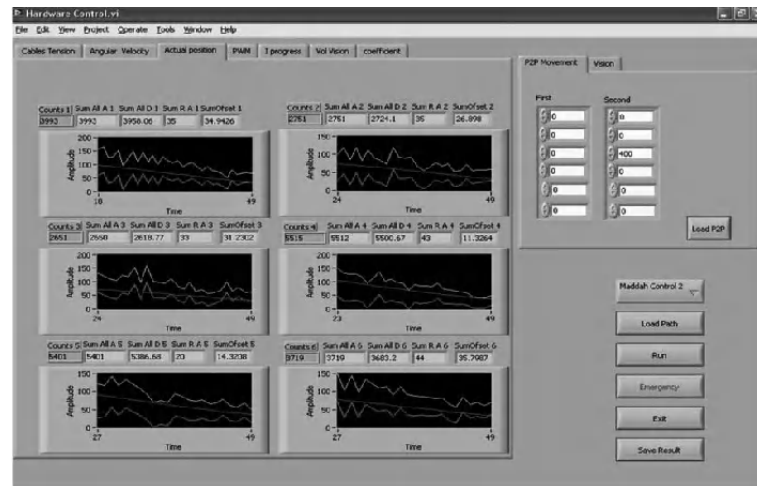


Figure 13.32. The graphical user interface (GUI) software setup programmed in LabVIEW environment

By using this software setup, a user is able to generate the plan of the robot's trajectory and run the robot to move in the related desired path. It is also possible to run each motor with a desired speed separately by producing its related PWM. The software setup sends the command signal to each motor by determining the pulse width of PWM signal by a data acquisition board. It also reads the output signals of encoders and calculates the position, orientation, linear velocity and angular velocity of the end-effector. It also plots the motion diagram and the position and angular velocity diagrams of each motor in a real-time way. In Figure 13.32, a panel of software is shown in which the desirable and the real positions of each six motors are being plotted simultaneously in a real-time way.

Since the whole GUI and simulator of the cable robot are designed in LabVIEW environment, it was required to transfer the analyzed data of imaging process and also lasers, which have been processed in Matlab[®] environment into the LabVIEW in an online way. Providing the controlling and timing consistency of these two softwares (Matlab[®] and LabVIEW) is not possible with the aid of a single PC. Therefore, two PCs are used here: the second PC analyzes the image processing of the camera data (Matlab[®]) while the first PC is responsible for processing the data of the lasers and also the process related to the dynamics and control of the robot in an online way.

An online timing was required to provide the consistency of these two PCs, which is done with the aid of methodology shown in Figure 13.33.

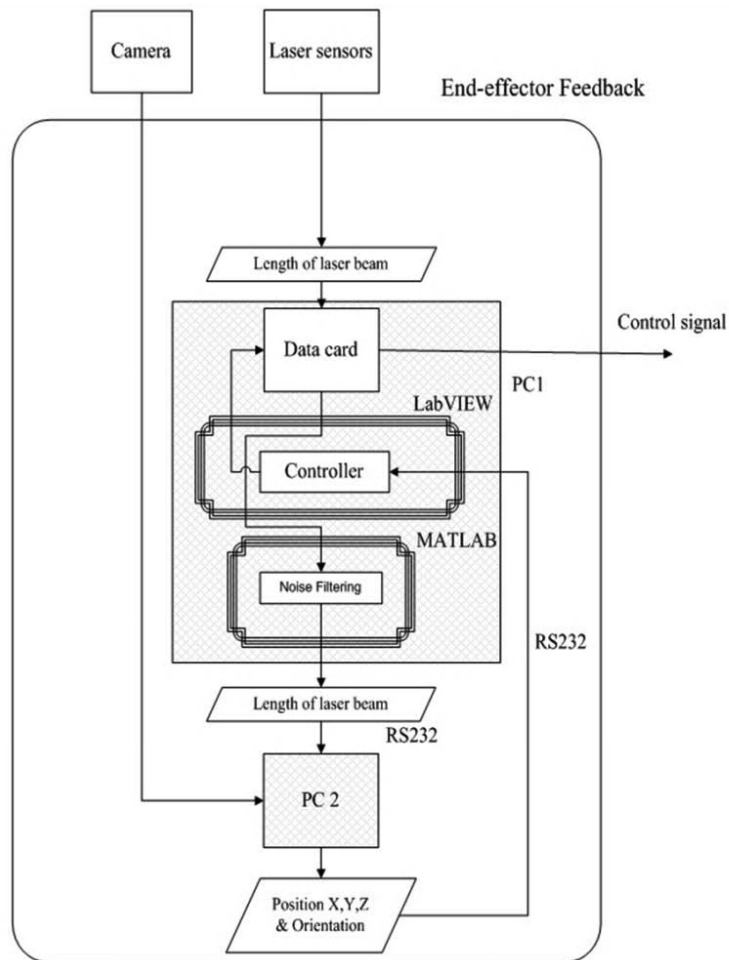


Figure 13.33. Flowchart of information architecture

First, the data related to the lasers are received by PC1 that is responsible for controlling the robot through the employed digital data card (A/D1780). Then, the data are transferred to the second PC with the aid of serial port and RS232 protocol that is responsible for analyzing the camera. Using the raw (virtual) data of image processing together with the altitude of the

end-effector, which is provided by the aid of lasers, it is possible to evaluate the actual position of the end-effector with the aid of perspective formulation in PC2. Finally, the obtained data will then be transferred into PC1 with the aid of the same serial protocol to be used as the end-effector feedback in controlling the robot.

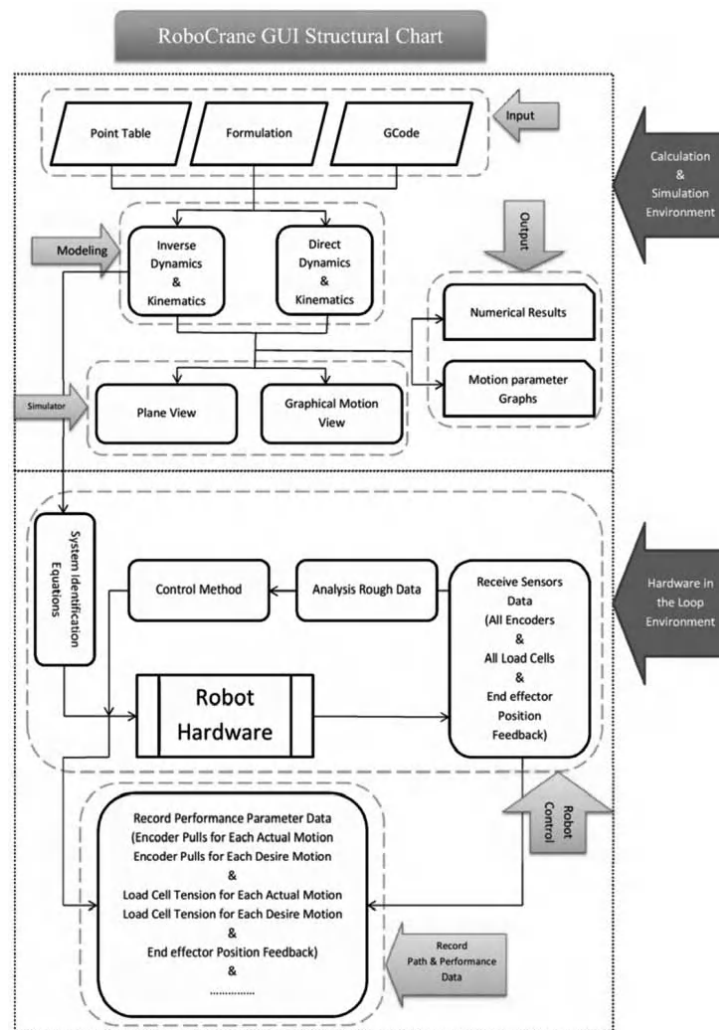


Figure 13.34. Flowchart of the structure of the designed GUI

13.6. Graphical user interface (GUI) [KOR 11d]

The interface is designed in a way to support these modules: taking the input; modeling, simulating and controlling the actuators; recording the sensors and saving the output. The following flowchart displays the overall scheme of the interface structure.

13.6.1. Simulator and virtual laboratory [KOR 11d]

The simulator interface of the ICaSbot is prepared in the LabVIEW, which is called RoboCrane, and is designed to be compatible with all versions of Windows. As soon as the mentioned interface is called, the main menu of this program appears in which the following options are available (Figure 13.35): Direct Kinematics, Inverse Kinematics, Direct Dynamics, PDV & Inverse Dynamics, Move Robot, Experiment, Safety Rules, Ports and Exit.

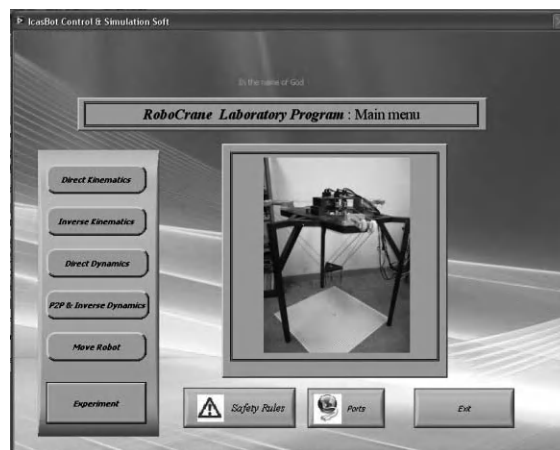


Figure 13.35. Main page of RoboCrane

Moreover, two selections of ports and safety keys are also positioned in the main menu. The ports key is prepared in order to make a connection to the robot using related interface electrical circuits. In this case, the robot can be easily controlled by the user through the designed graphical interface while all of the kinematics and dynamics results are available to be observed and interfered with. Finally, the safety rules key includes a series of warnings and primitive data that are necessary for the users to be considered.

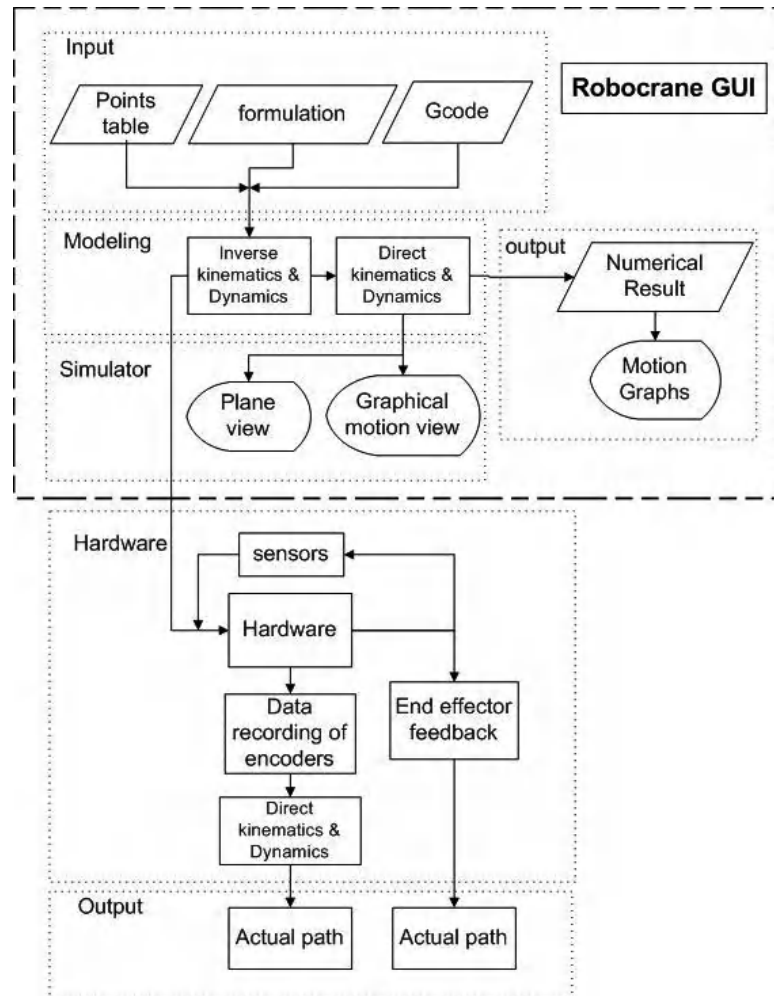


Figure 13.36. Flowchart of the structure of the designed GUI

In the direct kinematics window (Figure 13.37), the user is able to obtain the final position of the end-effector by specifying the amount of angular rotation of the motors together with the initial position of the end-effector. Also, the direct kinetics window (Figure 13.38) provides a virtual environment in which the user is able to estimate the actual position of the end-effector as the result of inserting a particular value of torques of the motors.

The 'Direct Kinematics_Enter Form.vi' window is divided into several sections:

- Tool Home Position:** Input fields for x (mm), y (mm), z (mm), S1e (Deg), Teta (Deg), and Phi (Deg), all set to 0.
- Workspace range:** A table of ranges for x, y, z, S1e, Teta, and Phi.
- Motor Rotation:** Input fields for Beta1 (rad) through Beta6 (rad), all set to 0.
- Workspace range:** A table of ranges for Beta1 through Beta6.
- Buttons:** 'Show Result', 'Example', 'Delta t = 0', 'Plan View', 'Reset', 'Isometric View', 'Move Robot', and 'Main menu'.

Tool Home Position	Workspace range	Motor Rotation	Workspace range
x (mm) 0	-300 < x < 300	Beta1 (rad) 0	-30 < Beta1 < 30
y (mm) 0	-300 < y < 300	Beta2 (rad) 0	-30 < Beta2 < 30
z (mm) 0	100 < z < 900	Beta3 (rad) 0	-30 < Beta3 < 30
S1e (Deg) 0	-45 < S1e < 45	Beta4 (rad) 0	-30 < Beta4 < 30
Teta (Deg) 0	-45 < Teta < 45	Beta5 (rad) 0	-30 < Beta5 < 30
Phi (Deg) 0	-45 < Phi < 45	Beta6 (rad) 0	-30 < Beta6 < 30

Figure 13.37. Direct kinematics window

The 'Direct Dynamics_Enter Form.vi' window is divided into several sections:

- Initial Position:** Input fields for X(mm), Y(mm), Z(mm), S1e(Deg), Teta(Deg), and Phi(Deg), all set to 0.
- Workspace range:** A table of ranges for X, Y, Z, S1e, Teta, and Phi.
- Motors Torque:** Input fields for Torque1 (Nm) through Torque6 (Nm), all set to 0.
- Torque range:** A table of ranges for Torque1 through Torque6.
- Buttons:** 'Example', 'Show Result', 'Isometric View', 'Plan View', 'Move Robot', and 'Main menu'.

Initial Position	Workspace range	Motors Torque	Torque range
X(mm) 0	-300 < x < 300	Torque1 (Nm) 0	-1.7 < Torque1 < 1.7
Y(mm) 0	-300 < y < 300	Torque2 (Nm) 0	-1.7 < Torque2 < 1.7
Z(mm) 0	100 < z < 900	Torque3 (Nm) 0	-1.7 < Torque3 < 1.7
S1e(Deg) 0	-45 < S1e < 45	Torque4 (Nm) 0	-1.7 < Torque4 < 1.7
Teta(Deg) 0	-45 < Teta < 45	Torque5 (Nm) 0	-1.7 < Torque5 < 1.7
Phi(Deg) 0	-45 < Phi < 45	Torque6 (Nm) 0	-1.7 < Torque6 < 1.7

Figure 13.38. Direct dynamic window

The required torque of each motor for a desired motion of the end-effector can be evaluated in the inverse kinetics (Figure 13.39) of this software by providing the desired path of the end-effector. The user is free to determine the desired path of the end-effector by providing a table of desired points or determining the equation of the desired path.

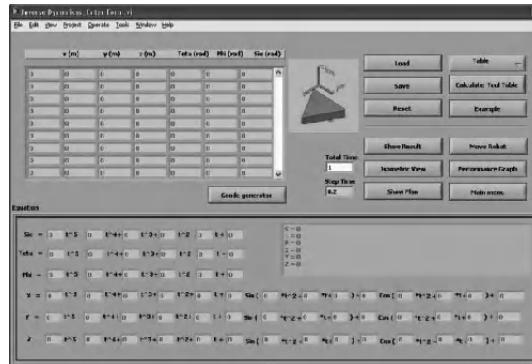


Figure 13.39. Inverse dynamic window

Finally, the traditional way of implementing the desired tracking of the end-effector, which is G-code, is also considered in the designed GUI, which is selected in the experiments window (Figure 13.40). The software is also able to provide the equivalent G-code of every motion of the end-effector from the input data of inverse kinetic window [KOR 11d].



Figure 13.40. G-code section

One of the most considerable facilities, which are designed in the proposed GUI, is the simulator section of the interface in which the user is able to observe the isometric motion of the robot in a virtual environment based on the input data.

Also the DoFs profile of the end-effector motion can be seen in another environment from different point of views in an online way for which the rate of simulating steps can be determined freely by the user (Figures 13.41 and 13.42).

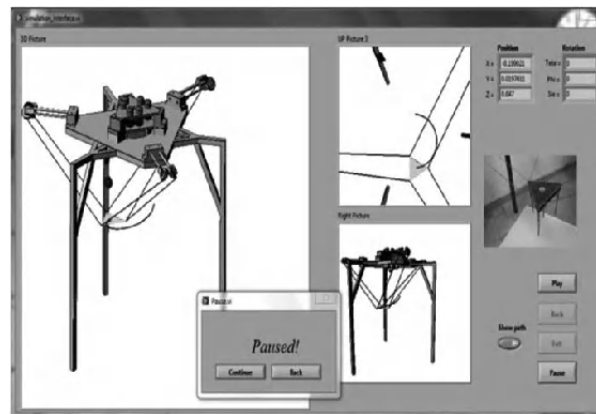


Figure 13.41. *Isometric view window*

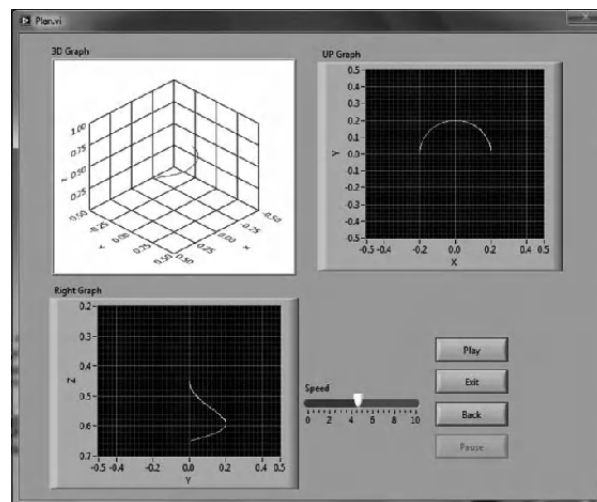


Figure 13.42. *Plane view window*

The user is also able to view the resultant profiles of the motors' torque, DoFs velocity and acceleration and cables' elongation as a function of time in this window (Figure 13.43).

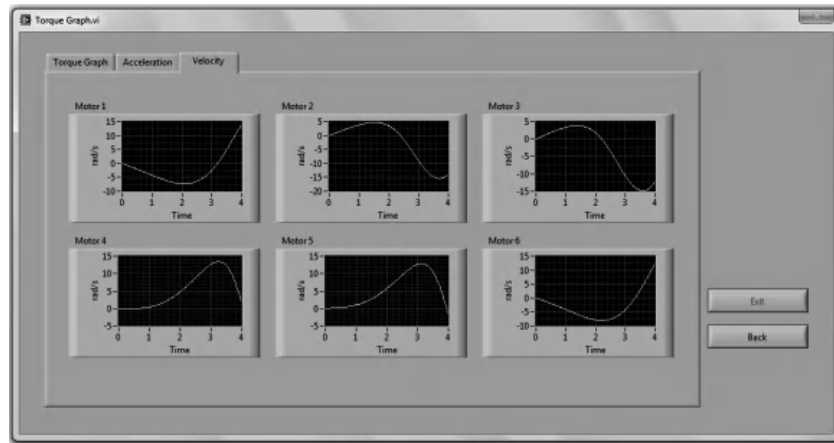


Figure 13.43. Motion graphs

13.6.2. Hardware control [KOR 11d]

Finally, the real robot can be controlled in the “hardware in the loop result” window. This part is the main control section of the software, which connects the robot to the electrical boards. The controlling commands that are driving the motors and also monitoring the actual data that are received from the sensors, can be managed in this part of GUI. Besides the control of the end-effector, the user is able to view and save the data related to the end-effector's motion including angular velocities of the motors, which are recorded by the encoders, cables' tension, which is recorded by the aid of load cells, and actual position of the end-effector, which is recorded by the position sensors.

In the main input window of the robot control section, by loading a predefined path, or by inputting the initial and final points (in point-to-point movement), the user commands the robot to move (Figure 13.44). Also, all the control methods explained in section 13.6.1 can be selected in this window. In addition, the communication parameters for the control of

hardware by computer, image processing and also the calibration coefficients of load cells and optical sensors are adjusted in this window. After the path is loaded and the control method is selected in the hardware control program, the angular velocity of motors and the required motors' torque are calculated from the inverse dynamic and kinematic equations and used as initial values for the adjustment of movement parameters.

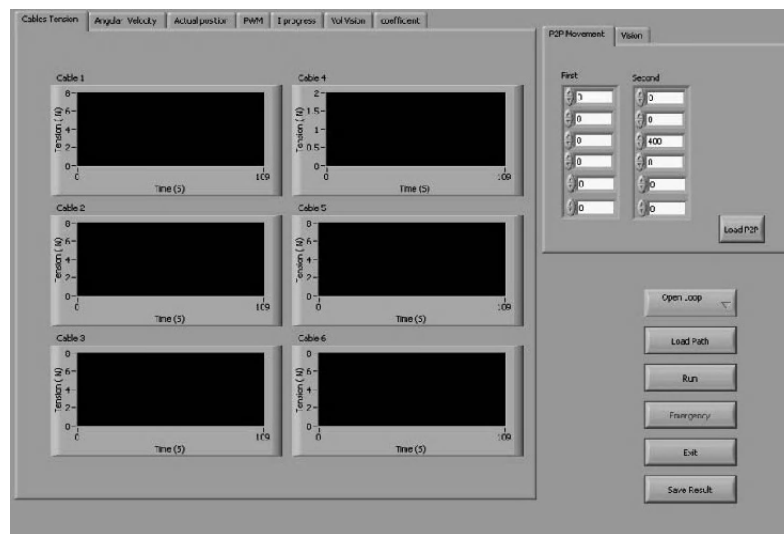


Figure 13.44. Main control window [KOR 11d]

After applying the considered inputs to the robot, all the received data from sensors like the encoder and load cell can be observed by going to the output window's tab. Also, the processing of images along with the robot movement can be observed and analyzed by referring to separate tab.

In the tab related to the load cell, the amount of voltage received from the load cell, which is detected through the analog-to-digital ports, can be observed.

In another tab titled “the ON time duration of motor control pulses”, the user can observe the ON time duration of pulses associated with all the motors and can evaluate the performance of control coefficients (Figure 13.46).

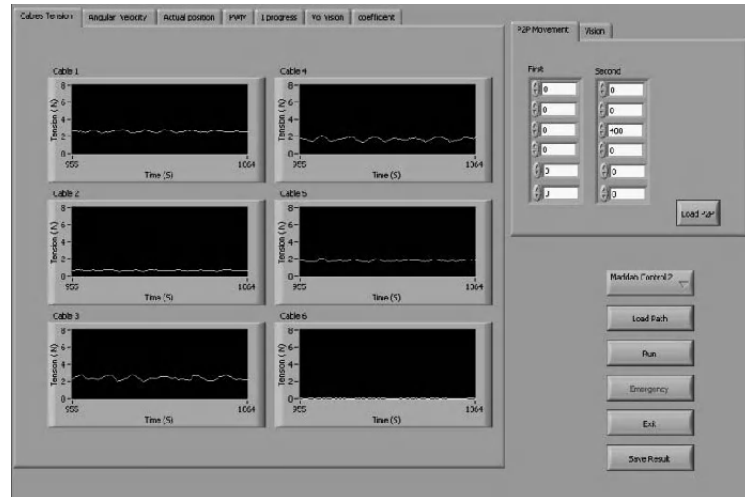


Figure 13.45. Load cells' tab [KOR 11d]



Figure 13.46. ON time duration of motor control pulses' tab [KOR 11d]

In the next tab, the data related to real pulses received and the ideal pulses that are needed and also the differences between these two are displayed. A larger difference means a larger deviation in path tracking. Also in this window, the total quantity of desired pulses that should be received during

tracking and also the total quantity of real pulses received during tracking are illustrated, which is indicative of the error in the end point (Figure 13.47).

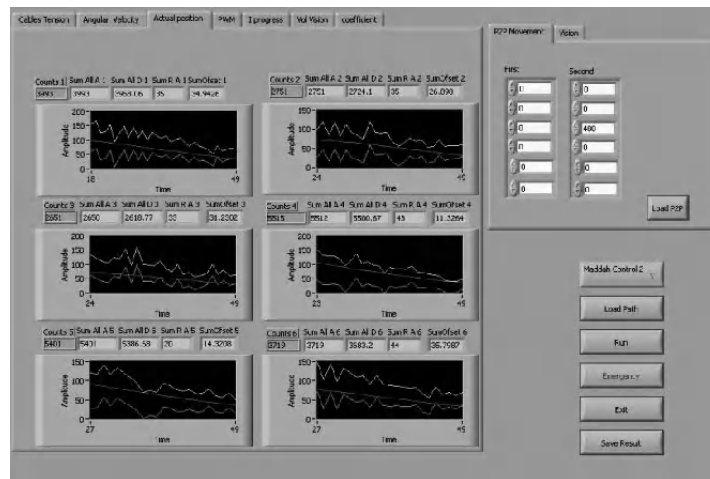


Figure 13.47. Received pulses' tab [KOR 11d]

In the tab related to the end-effector, the user can observe the data of the optical sensors, which are simultaneously being executed by an image processing program on a second computer (Figure 13.48).

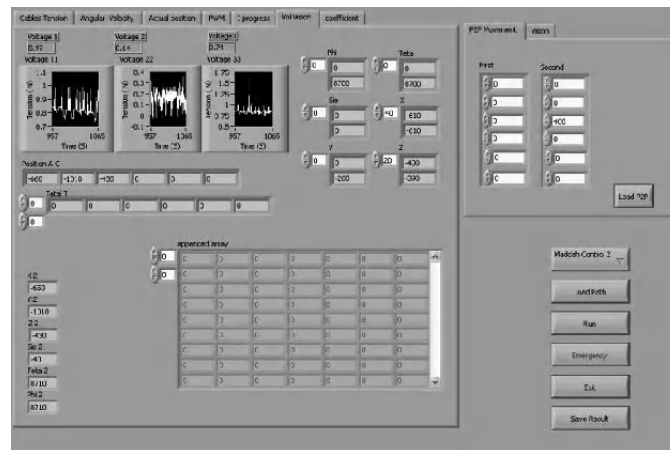


Figure 13.48. The end-effector data's tab [KOR 11d]

Moreover, the user can save the path-related data. These data include the load cell data, encoder data and the data received from the end-effector position.

The result of trajectory tracking in virtual simulator and also the camera data are shown in Figures 13.49 and 13.50. It can be seen from the experimental result that an acceptable compatibility can be observed between the simulation results and experimental test.

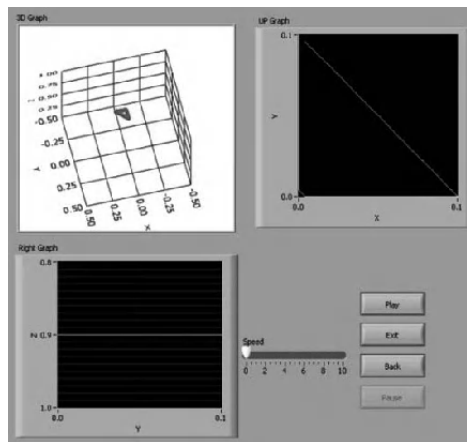


Figure 13.49. Isometric view and plan view of triangle tracking in GUI simulator [KOR 11d]



Figure 13.50. Result of trajectory tracking in virtual simulator [KOR 11d]

13.7. Result and verifications

There are some ISO tests defined in the references, which evaluate the performance of the robot maneuverability. One of the most powerful tests of this type is ISO 9283 for which a trajectory should be tracked by the robot on a plane inclined by a 45° angle and its norm of deviation error shows the performance of the robot [KOR 11c]. The ISO test that is considered here for verifying the precision of the robot is a 45° inclined circle with the following equation:

$$\begin{aligned} x &= 0.05 \cos(0.14\pi(5.37 - t)^2) \\ y &= -0.05 \cos\left(\frac{\pi}{4}\right) \sin(0.14\pi(5.37 - t)^2) \\ z &= -0.05 \sin(\pi/4) \sin(0.14\pi(5.37 - t)^2) + 0.8 \cos(\pi/4) \end{aligned} \quad [13.29]$$

where t is time. Through this test, the tracking performance of the ICaSbot, which is equipped with the mentioned closed-loop controller, is verified and we can compare the efficiency of the two mentioned controlling strategies from the dynamic error point of view.

For the whole simulation results and profiles, comparison is done between the controlling strategy based on the new proposed feedback recording (Cartesian space control) and the previous strategy that has been used so far for all of the cable robots, which is using joint space control or control based on the feedback of the encoders to show the efficiency of the proposed method. First, it can be observed that there is an acceptable compatibility between the experimental tests and simulation results in which the error is less than 1 cm. As can be seen in Figure 13.51, the tracking accuracy is considerably better for the robot that is equipped with the end-effector feedback controller rather than the robot that is controlled by the aid of joint space controller. For the end-effector controller, there is a little deviation at the first stage of movement, which is compensated over the time with the aid of end-effector controller as was expected. The elliptic shape of the x - y profile is due to altitude variations of the end-effector whilst tracking an inclined circle. A comparison of angular velocity of the motors among simulation, experimental results of joint space controller and experimental results of Cartesian space controller are shown in Figure 13.52.

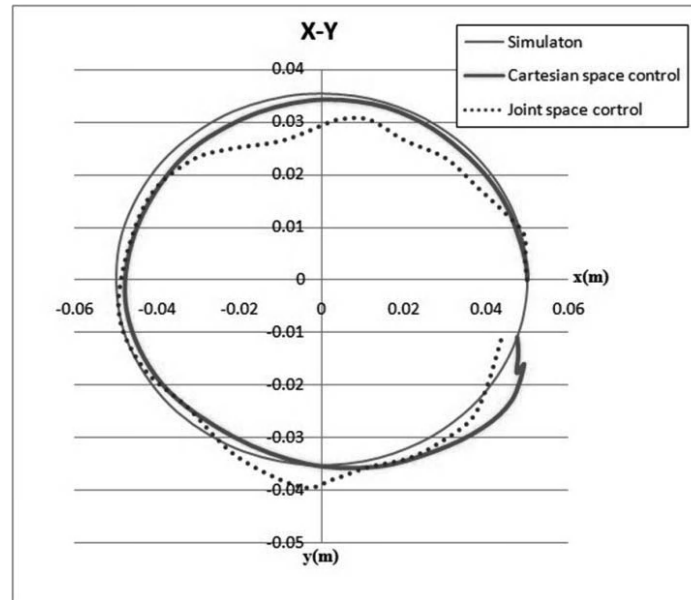


Figure 13.51. Comparing the result of simulation path and experiment path for inclined circle

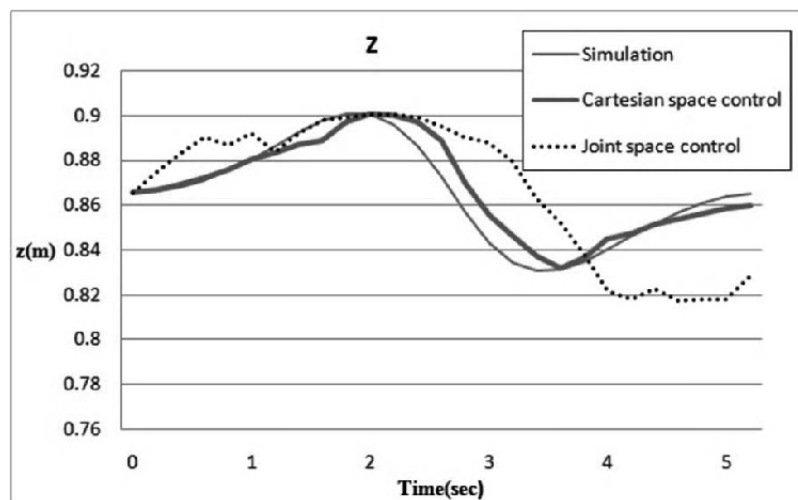


Figure 13.52. Comparing the result of simulation path and experiment path for inclined circle

It can be seen that as a result of the high inertia of the motors, the experimental profiles have a delay compared to simulation profiles at the initial moments of movements; however, the compatibility increases afterwards, thanks to implementing the designed controller. As a result, the overall area under the angular velocity curve of the motor, which specifies the overall rotation of the motors during the time, is roughly equal for simulation and experimental results. Again, it can be observed that this compensation and the compatibility of angular velocity profiles are extremely better in the case of the Cartesian space controller compared to the joint space controller.

In order to have a more understandable comparison of the results, the norm of error function, which can be calculated by the following equation, is plotted for two controlling cases i.e. joint space control and Cartesian space control:

$$E = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad [13.30]$$

In this equation, x_1, y_1, z_1 is the desired position and x_2, y_2, z_2 is the actual position of the end-effector in trajectory tracking.

As we expected, the normal error of the Cartesian space control method is less than the joint space control method and has the average error of approximately 1 cm versus 2 cm in joint space control mode. Cable tension results and its comparison with simulation are shown in Figure 13.54.

As can be seen from diagrams, at some points cable tension value is deviated from the simulation value, which is due to the existence of the extra load of the end-effector and its effect on the friction and inertia of the system that causes the experimental results to be higher than simulation values. Moreover, the deviations are compensated quickly by the designed control system. As was mentioned for the previous test, it can be seen from the obtained results of most load sensors that the robot endures higher tension outputs compared to simulation at the initial stages of the motion, which is due to the high applied force on cables to overcome friction and to initiate the motion in the beginning. This phenomenon is more serious in this test compared to the previous test since the robot moves with its maximum load capacity. Afterwards, the robot tolerates lower cable tension in the next few seconds by a quick compensation of deviations. However, again here, the mean value of cable tensions of experimental motion is roughly equal to the

simulation value, which validates the correctness of the experiment. Moreover, like the previous observations, there are some oscillations in the practical diagram compared to the simulator diagram in all of the obtained results. The reasons can be contributed to the following facts: because of using PD control along trajectory in motors, the creation of oscillation in robotic motion is inevitable. This oscillation has effects on cable tensions. Flexibility of the joints, cables and structure are other sources of this vibrating response. These flexibilities of the cables and also the structure of the robots affect the vibrating response of the robot in reality. Clearance, especially in the joints, motors, drum, pulleys, flip flop, cable tension measurement mechanism and cable conducting pulley, is also highly effective in the creation of oscillation in the cable tensions' diagrams during the motion.

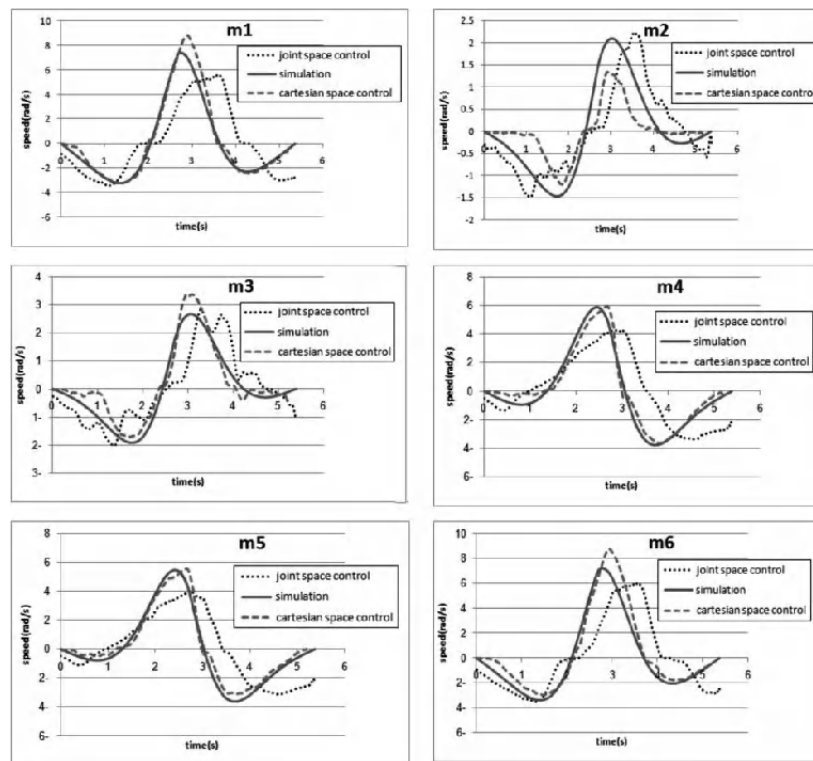


Figure 13.53. Comparing the simulation and experiment angular velocity of motors for inclined circle

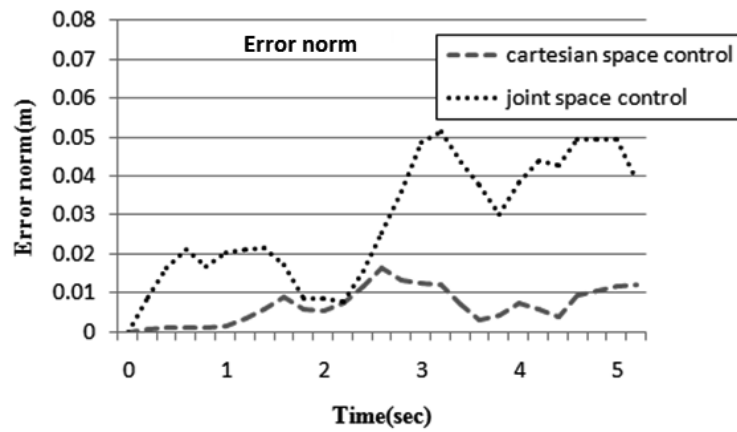


Figure 13.54. Error norm in inclined circle path

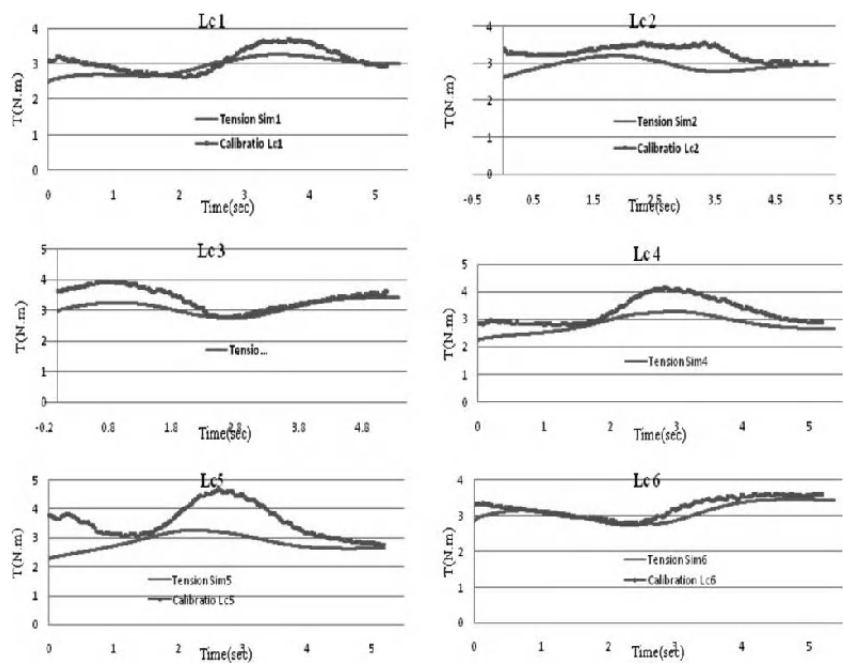


Figure 13.55. Comparing the cables' tension between simulation results and experimental data

13.8. Conclusion

As has been explained, parallel robots are widely used in industrial environments and on automation lines as a result of their high stiffness, load capability and acceptable accuracy. The parallel nature of these robots and necessity of providing a powerful real-time and online control for them converts these robots to a complicated mechatronics mechanism. It was stated that one of the most applicable kinds of these robots, which has been developed recently, is cable robots in which the pneumatic and hydraulic jacks are replaced by cable and motor actuators. High ratio of load to weight, large workspace and perfect capability of assembling and de-assembling are the positive consequences of this strategic replacement. On the other hand, its parallel dynamic formulation and the tensional force limitation of these kinds of robots make their control more challenging.

In this chapter, after a brief introduction of the overall structure of these robots and their applications, required dynamic and controlling formulation of these robots have been represented. It was seen that a proper nonlinear control strategy is required to control the end-effector on a predefined trajectory. This control consists of a feed-forward term in which the whole inverse kinematics and kinetic formulations are employed. This feed-forward term is then improved using feedback signals with the aid of two sequential closed-loop controllers. The inner controller controls the angular velocity of the motor for each 0.01 s while the end-effector feedback is used in the outer closed loop for each 0.1 s.

It was stated that this complicated controlling strategy needs high-tech mechatronics equipment and hardware installation including powerful actuators, high-resolution sensors, compatible data transfer protocol and a user-friendly simulator with GUI [HAB 06, HAB 07, HAB 08]. Six DC motors together with six cables rolled up with the aid of six pulleys are used as the required actuator. It was seen that with the aid of providing a suitable lookup table, it is possible to determine the required torque and angular velocity using the employed DC motors that can be used to control the end-effector based on inverse dynamics. Its related electrical boards and drivers were designed, and required calibration and a lookup table were provided. Six encoders are used to feed back the actual angular velocity of the motors. In order to estimate the actual tension of the cables, six load cells are installed. Again, required electrical interface boards and their related calibration were presented. It was seen that in order to sense and feed back all of the translational and rotational movements of the end-effector, it is required to use a combination of lasers and a camera in which the lasers help to evaluate the altitude of the end-effector together with its angles while the camera helps us

to estimate the planar coordinates of the end-effector. Data acquisition cards were used to provide a high-speed transfer protocol to determine the required real-time and online heavy calculations of the parallel robot. Digital cards are a help to manage the motor and encoder data while the analog card is used to handle the load cell data. Finally, in order to train the operator and also control the robot, a simulator and GUI are designed in the LabVIEW environment by which the desired commands can be implemented on the robot using different ways and all of the resultant actual kinematics and kinetics data can be monitored. In the end, in order to validate the mentioned dynamics and control strategies and also to verify the efficiency of all the mentioned mechatronics implementation of the robot including its hardware and software setup, a comparison was performed between the simulation result of Matlab[®] and experimental data. Not only was a good compatibility shown, but also complete analyses were delivered from dynamic and control points of view.

It can be seen that all of the requirements of a complete mechatronics mechanism are involved in this parallel robot and the importance of mechatronics knowledge is shown in this chapter regards design, manufacture and running the above-mentioned robot with good accuracy.

Definition	Symbol
Translational coordinates of the end-effector	$x_m, y_m, z_m (m)$
Rotational coordinates of the end-effector	$\psi, \theta, \phi (\text{rad})$
Jacobian matrix	J
Cables' elongation	L
Pulleys' angular rotation	β
Pulley radius	R
Vector of cables tension	T
Inertia matrix	$D(x)$
Vector of velocity terms	$C(x, \dot{x})$
Gravity vector	$g(x)$
Vector of DoFs of the system	X
Mass of the end-effector	M
Moment of inertia of the end-effector	I
Length of the cables	Q
Matrix of rotary inertia of the motors	J
Viscous friction matrix of the motors	C
Vector of angular velocity	$\dot{\beta}$

Table 13.8. Specification of the used optical laser sensor

Vector of motors torque	τ
Control input of the feedback linearization	v
Controlling gains of derivative errors	K_{iD}
Controlling gains of proportional errors	K_{iP}
Desired trajectory of the robot	X_d
Actual trajectory of the robot	X_a
Armature voltage of the DC motor	v_a
Resistance of armature wire of the DC motor	R_a
Armature inductance of the DC motor	L_a
Armature current of the DC motor	i_a
Emf (induced voltage of electromotor force)	e_a
Constant of torque of the DC motor	K_m
Produced torque of the DC motor	τ_m
Angular velocity of the DC motor	ω_m
Maximum voltage of the motor	v_{max}
Controlling gains of proportional error of motor angular velocity	K_d
Controlling gains of derivative error of motor angular velocity	K_p
Controlling gains of proportional error of motor torque	K'_d
Controlling gains of derivative error of motor torque	K'_p
Distance between the camera and the point	H
Canonical distance of camera	F
Local coordinate system	X, Y, Z
Angle between the direction of the laser beams and local Z axis	ζ
Angle of normal vector of the end-effector	γ
Global coordinate system	$\hat{X}, \hat{Y}, \hat{Z}$

Table 13.8. (Continued) Specification of the used optical laser sensor

13.9. Bibliography

- [ALB 93] ALBUS J., BOSTELMAN R., DAGALAKIS N., "The NIST ROBOCRANE", *Journal of Robotics Systems*, vol. 10, no. 5, pp. 709–724, 1993.
- [ALP 01] ALP A.B, Cable-suspended parallel robots, MS Thesis University of Delaware, 2001.

- [BOS 95] BOSTELMAN R., ALBUS J., DAGALAKIS N., *et al.*, *RoboCrane Project: An Advanced Concept for Large Scale Manufacturing*, Intelligent Systems Division National Institute of Standards and Technology Gaithersburg, MD, 1995.
- [CHA 04] CHANG A.H., Experimental development of the mobile vestibular platform, SUNFEST Technical Report, Center for Sensor Technologies, University of Pennsylvania, Philadelphia, PA, 2004.
- [HAB 06] HABIB M.K., “Mechatronics engineering the evolution, the needs and the challenges”, *Proceedings of the 32nd Annual Conference of IEEE Industrial Electronics Society*, IECON 2006, Paris, pp. 4510–4515, 2006.
- [HAB 07] HABIB M.K., “Mechatronics: a unifying interdisciplinary and intelligent engineering paradigm”, *IEEE Industrial Electronics Magazine*, vol. 1, no. 2, pp. 12–24, Summer 2007.
- [HAB 08] HABIB M.K., “Interdisciplinary mechatronics: problem solving, creative thinking and concurrent design synergy”, *International Journal of Mechatronics and Manufacturing Systems (IJMMS)*, vol.1, no.1, pp. 264–269, 2008.
- [IRA 09] IRANPOUR M., Modeling and simulation of kinematics and dynamics of spatial cable-actuated-robots, MSc Thesis, Iran University of Science and Technology, 2009.
- [KAM 95] KAMAMURA S., CHOE W., TANAKA S., *et al.*, “Development of an ultrahigh speed robot FALCON using wire drive system”, *IEEE International Conference on Robotics and Automation*, Japan, 21–27 May 1995.
- [KOR 09] KORAYEM M.H., BAMDAD M., “Dynamic load carrying capacity of cable-suspended parallel manipulators”, *International Journal of Advanced Manufacturing Technology*, vol. 44, pp. 829–840, 2009.
- [KOR 10a] KORAYEM M.H., TOURAJIZADEH H., “Maximum DLCC of spatial cable robot for a predefined trajectory within the workspace using closed loop optimal control approach”, *Journal of Intelligent & Robotic Systems*, 2010.
- [KOR 10b] KORAYEM M.H., TOURAJIZADEH H., BAMDAD M., “Dynamic load carrying capacity of flexible cable suspended robot: robust feedback linearization control approach”, *Journal of Intelligent and Robotic Systems*, vol. 60, no. 3–4, pp. 341–363, 2010.
- [KOR 11a] KORAYEM M.H., BAMDAD M., TOURAJIZADEH H., *et al.*, “Development of ICaSbot: a cable-suspended robot’s with six DOF”, *Arabian Journal for Science and Engineering*, 2011.
- [KOR 11b] KORAYEM M.H., JALALI M., TOURAJIZADEH H., “Optimal path planning of spatial cable-suspended robot subject to maximum load carrying capacity using optimal sliding mode control approach”, *3rd International Conference on Manufacturing Engineering ICME2011*, Tehran, Iran, 2011.

- [KOR 11c] KORAYEM M.H., KHAYATZADEH S., SAFFARI A.H., *et al.*, “Sensing the position of the end-effector of a cable suspended robot for on-line control”, *3rd International and 12th National Conference on Manufacturing Engineering (ICME)*, University of Tehran, Iran, 2011.
- [KOR 11d] KORAYEM M.H., TAHERIFAR M., MADDAH S.M., *et al.*, “Design and programming a graphical user interface for the ICaSbot Robot using LabVIEW”, *International Conference on Computational Intelligence Applications (ICCIA)*, Shiraz, Iran, 2011.
- [KOR 12a] KORAYEM M.H., NAJAFABADI A.I., TOURAJIZADEH H., “Control of ICaSbot cable robot using simultaneous control of the speed and torque of the motors”, *International Conference on Experimental Solid Mechanics and Dynamics (X-MECH)*, Iran University of Science and Technology, Tehran, Iran, 2012.
- [KOR 12b] KORAYEM M.H., TAJIK A., NAJAFABADI A.I., *et al.*, “Calculation of the end-effector position of the cable robot using cable tension and encoders outputs”, *3rd International and 12th National Conference on Manufacturing Engineering (ICME)*, University of Tehran, Iran, 2012.
- [KOR 12c] KORAYEM M.H., TOURAJIZADEH H., JALALI M., *et al.*, “Optimal path planning of spatial cable robot using optimal sliding mode control”, *International Journal of Advanced Robotic Systems*, INTECH, 2012.
- [OH 05] OH S.R., AGRAWAL S.K., “A reference governor-based controller for a cable robot under input constraints”, *IEEE Transactions on Control Systems Technology*, vol. 13, pp. 639–645, 2005.
- [OSU 93] OSUMI H., ARAI T., ASAMA H., “Development of a seven degrees of freedom crane with three wires (1st Report) – concept, design and control”, *The Japan Society of Precision Engineering*, vol. 59, no. 5, pp. 767–772, 1993.
- [ROS 06] ROSATI G., GALLINA P., ROSSI A., *et al.*, “Wire-based robots for upper-limb rehabilitation”, *International Journal of Assistive Robotics and Mechatronics*, vol. 7, pp. 3–10, 2006.
- [ROS 07] ROSATI G., GALLINA P., MASIERO S., “Design, Implementation and clinical tests of a wire-based robot for neurorehabilitation”, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 15, no. 4, pp. 560–569, 2007.
- [SHI 99] SHIANG W.J., CANNON D., GORMAN J., “Dynamic analysis of the cable array robotic crane”, *IEEE International Conference on Robotics and Automation*, Detroit, MI, pp. 495–500, 1999.
- [TAJ 11] TAJIK A., KORAYEM M.H., TOURAJIZADEH H., “Developing a force measurement system for cable suspended robot and study its applications”, *3rd International and 12th National Conference on Manufacturing Engineering (ICME)*, University of Tehran, Iran, 2011.

- [VAR 06] VARZIRI M.S., NOTASH L., “Kinematic calibration of a wire-actuated parallel robot”, *Mechanism and Machine Theory*, vol. 42, no. 8, pp. 960–976, August 2007.
- [YAM 99] YAMAMOTO M., YANAI N., MOHRI A., “Inverse dynamics and control of crane-type manipulator”, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Kyongju, South Korea, pp. 1228–1233, 1999.

Chapter 14

A Localization System for Mobile Robot Using Scanning Laser and Ultrasonic Measurement

14.1. Introduction

The ability of locating itself in the environment is essential for a mobile robot to execute its commands. Localization is fundamental to subsequent tasks such as map building and collision-free path planning. Various methods have been developed by many researchers. A review on the mobile robot localization technologies can be found in [BOR 97]. Among those, odometry was widely used to determine the momentary position. Borenstein and Feng [BOR 96] improved the accuracy of odometry up to one order of magnitude by measuring and correcting the robot's systematic errors. However, wheel slippage and ground roughness would cause a fatal error in the robot's heading measurement, which could later introduce a significant position error, should the robot travel a long way. In order to improve the robot's heading accuracy as well as the dead reckoning accuracy, gyroscope was used and tested by many researchers [KOM 94, HAR 96, FEN 96, BOR 98, CHU 01]. The dead reckoning method using a gyroscope can avoid the influence of wheel slippage and ground roughness on a robot heading measurement, and hence provides a more accurate momentary position and heading than odometry does. But low-cost gyroscopes suffers from random

Chapter written by Kai LIU, Hongbo LI and Zengqi SUN.

drift over long-time running. Dead reckoning is a relative localization method, and may suffer from unbounded accumulative errors. To compensate this, an absolute method must be used. Triangulation using landmarks is a commonly adopted method [COH 92, BET 97, FON 05]. To use artificial landmark triangulation, the position of each landmark must be known. This is not convenient when the robot's working environment is unstructured. To use natural landmark triangulation, an accurate map for the environment should be available. Often, to build a map of the robot's environment is a tough task for users. Tsai [TSA 98] combined a flux gate compass, a rate-gyro and two encoders to get an optimal estimate for robot heading, and combined the dead reckoning position estimate with ultrasonic range measurements to remove accumulated position errors, but the coverage of the ultrasonic transmitter was limited to a small area. And this method fails in the environment where the magnetic field of the earth is twisted. Vision-based localization and simultaneous localization and mapping (SLAM) are regarded as key essentials for a real autonomous mobile robot and have been widely researched in recent years [SE 05, ROY 07]. But the computational complexity hinders their realistic application.

Our efforts focus on developing a simple, low-cost, continuously accurate and convenient method of mobile robot localization using a multisensor. We assume that the mobile robot has a base station fixed to the ground where the robot can recharge itself. The robot's position and heading are relative to the base station. The method's novelty lies in the combination of scanning laser and ultrasonic measurement to construct an absolute localization system. A scanning laser and an ultrasonic transmitter are mounted on the base station to measure the robot's angle and distance relative to it. Another scanning laser is mounted on the robot to measure the base station's relative angle in the robot's frame. The base station and the robot are wirelessly connected through radio frequency (RF). Using the extended Kalman filter, the data of a scanning laser and ultrasonic absolute positioning system are fused with the dead reckoning system that utilizes a low-cost gyroscope. This method achieves the convenience that neither an extra artificial landmark nor an accurate map of the environment is needed, and it is suitable for both a structured and an unstructured environment.

The chapter is organized as follows. Section 14.2 describes the mechanical and electronic hardware configuration of our positioning system. Section 14.3 details the proposed method. The performance of the method is given in section 14.4. Section 14.5 presents our conclusion.

14.2. System configuration

The mechanical structure of our mobile robot is typical, which consists of two independently driven back wheels and a freely rotating front wheel. The robot has a digital signal processing (DSP) unit as its central controller. A low-cost and high drifting numerical Micro-Electro-Mechanical Systems (MEMS) gyroscope ADIS16255 is utilized in dead reckoning. Two optical incremental encoders are mounted on the driving motors to provide the odometry information of each wheel for the dead reckoning. A compass is not used for the reason that in some environments the magnetic field is twisted and it will lose its function or bring error heading information to the robot, especially when the magnetic field's direction changes slowly from place to place.

Because of the wheel slippage, surface roughness and gyro-drift, there is accumulative error in the robot position estimation. So, an absolute positioning system is needed to compensate the accumulative error in the dead reckoning system. To improve the convenience of installation, we make full use of the base station and no extra landmark is needed. According to polar coordinates system, to locate the robot, the angle and distance of the robot in the base station polar coordinates system should be known. A laser scanner with a high-resolution optical encoder is mounted on the base station to get the polar angle. An ultrasonic wave transmitter is attached to the scanning laser and rotates with it. Figure 14.1 shows the structure.

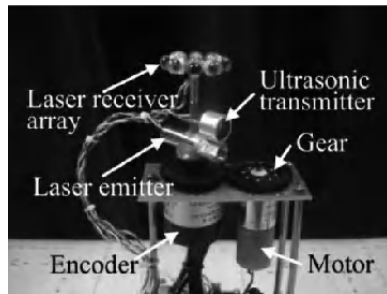


Figure 14.1. *Assembling structure of the scanning laser, ultrasonic transmitter and the laser receiver array on base station*

A laser receiver array and an ultrasonic wave receiver array are mounted on the robot. They can receive the laser and ultrasonic wave that comes from any direction from the base station. Figure 14.2 shows the structure.

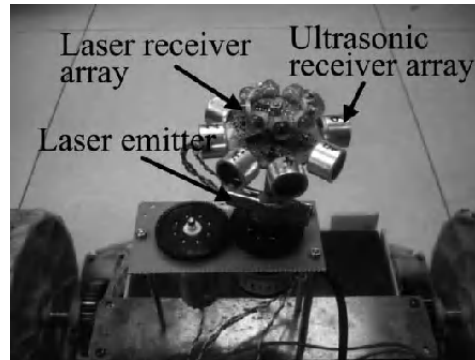


Figure 14.2. *Assembling structure of the laser receiver array, ultrasonic receiver array and the laser emitter on robot*

When the laser is received by the robot, it sends an RF signal to the base station to indicate that the robot is scanned by the base. Then, the base station sends the angle of the scanning laser encoder to the robot and emits an ultrasonic wave pulse after that. Since the time delay from the RF transmitting to the ultrasonic wave sending is constant, the time of flight (TOF) of the ultrasonic wave can be measured so as to obtain the distance from the base station to the robot.

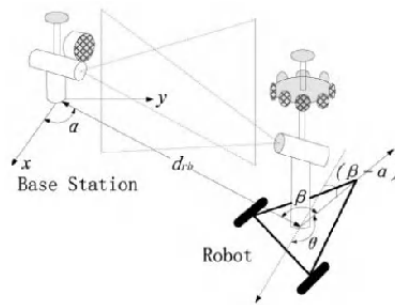


Figure 14.3. *Sketch of the scanning laser and ultrasonic absolute positioning system*

To compensate the angle drift of the low-cost gyroscope, we mounted another scanning laser on the robot, and the laser's direction is also measured by a high-resolution optical incremental encoder, as shown in Figure 14.2. The laser line emitted by the base station and the robot are modulated to

different frequencies to avoid mutual and sunlight interruption. The base station also has a laser receiver array to receive the laser from the robot. When the robot laser is received by the base station, it sends an RF signal to the robot. The moment robot receives the signal from the base station, the angle of the scanning laser encoder is read out. Together with the position of the base station and the robot, the robot heading can be calculated. And then fused with the gyroscope information, the optimal robot heading is estimated.

A sketch of the scanning laser and ultrasonic absolute positioning system is shown in Figure 14.3.

14.3. Implementation

14.3.1. *Dead reckoning system*

When the robot is not scanned by the base station, it works in a dead reckoning mode. Tsai [TSA 98] used a multisensorial scheme fusing data from a gyro, a flux compass and two encoders to get an optimal estimate of the robot heading. But when we apply this method, it fails when the direction of the Earth's magnetic field changes slowly from place to place. In this condition, the absolute difference between the compass heading and gyro heading is less than the selected threshold in every sampling period (0.5 s), and the error information in a compass is fused in the robot heading. Gradually, the robot heading converges to the direction of the magnetic field, which has been twisted. This is because the error model in a compass measurement is not a discrete-time, zero-mean white Gaussian process when the magnetic field is slowly twisted. So in our system, we measure the robot relative heading by fusing information from a gyro and two encoders, and later fuse the relative heading with scanning laser angle information when the base station is scanned by the robot. Let $\theta_e(k)$ and $\theta_e(k-1)$ denote the robot heading angle estimated by using the information from two wheel encoders at time k and $(k-1)$, so $\theta_e(k)$ can be calculated by:

$$\theta_e(k) = \theta_e(k-1) + (\Delta p_r c_r - \Delta p_l c_l) / d + n_e(k) \quad [14.1]$$

where Δp_r and Δp_l denote the pulse increase of right encoder and left encoder from time $(k-1)$ to time k . d denotes the distance between the driven wheels. c_r and c_l convert the encoder pulse to distance traveled by the right and left wheels in millimeters. In order to get the accurate value of c_r and c_l , we manually drive the robot to track along a straight line path for a

distance l , and read the pulse increase p_r and p_l . The c_r and c_l can be calculated by $c_r = l/p_r$ and $c_r = l/p_l$. Several measurements are needed to increase accuracy by calculating the average value of c_r and c_l . Measurement noise $n_e(k)$ is assumed to be a zero-mean white Gaussian process with variance σ_e^2 .

Let $\theta_g(k)$ and $\theta_g(k-1)$ denote the robot heading angle estimated from a gyroscope. For convenience, we rewrite here the equation of the gyro heading measurement in [TSA 98]:

$$\theta_g(k) = \theta_g(k-1) + \int_{(k-1)T}^{kT} \dot{\theta}(t) dt + n_g(k) \quad [14.2]$$

where $\theta(t)$ is the gyro's rotational rate. T is the integration period. The noise $n_g(k)$ is also assumed to be a zero-mean white Gaussian process with variance σ_g^2 . According to [TSA 98], if the absolute difference between $\theta_e(k)$ and $\theta_g(k)$ is less than a selected threshold, the dead reckoning optimal robot heading estimated by fusing information from the gyro and two encoders is:

$$\theta_d(k) = \frac{\theta_e(k)\sigma_g^2 + \theta_g(k)\sigma_e^2}{\sigma_g^2 + \sigma_e^2} \quad [14.3]$$

otherwise, $\theta_d(k) = \theta_g(k)$. After the fusing, we let $\theta_e(k) = \theta_g(k) = \theta_d(k)$.

The gyro and encoder information updating period that we use is 10 ms to provide the robot real-time heading information that is needed in its real-time control. $\theta_e(k)$ and $\theta_g(k)$ are fused every 0.5 s. We call it the fusing period. In between the fusing period, $\theta_d(k)$ is equal to $\theta_g(k)$. If the fusing period is too small, the error between $\theta_e(k)$ and $\theta_g(k)$ will be constantly below the threshold even when wheel slippage occurs. Therefore, error information will be fused in the robot heading.

Next, the robot's dead reckoned position (x_d, y_d) is updated every 10 ms by:

$$x_d(k) = x_d(k-1) + \Delta x_d \quad [14.4]$$

$$y_d(k) = y_d(k-1) + \Delta y_d \quad [14.5]$$

where

$$\Delta x_d = \frac{\Delta p_r c_r + \Delta p_l c_l}{2} \cos\left(\frac{\theta_d(k) + \theta_d(k-1)}{2}\right) \quad [14.6]$$

$$\Delta y_d = \frac{\Delta p_r c_r + \Delta p_l c_l}{2} \sin\left(\frac{\theta_d(k) + \theta_d(k-1)}{2}\right) \quad [14.7]$$

14.3.2. Scanning laser and ultrasonic positioning system

The dead reckoning positioning system is relative and will suffer from unbounded accumulative error. Therefore, an absolute positioning system is needed to eliminate the accumulative error in the robot heading and position in time. The scanning laser mounted on the base station scans the robot repeatedly. When the robot is scanned by the base station, it sends an RF signal. When the base station receives the RF signal, it sends the scanning laser angle α (shown in Figure 14.3) to the robot also through RF and then emits an ultrasonic wave pulse immediately. Let t_{rf} denote the moment when the RF (which contains the scanning laser angle) is received by the robot, and t_{ul} denote the moment when ultrasonic wave pulse is received by the robot. The radio signal travels in the air at light speed. So its flying time can be ignored, then the ultrasonic flying time is:

$$t_f = t_{ul} - t_{rf} - t_0 \quad [14.8]$$

where t_0 denotes the system's constant time delay from the RF transmitting to the ultrasonic wave sending in the base station and the RF receiving delay in the robot. As the sound speed is related to the density of air, there is a temperature sensor on the robot to compensate the sound speed variance. So, the distance between the robot and the base station, d_{rb} , is measured. Along with the angle of the scanning laser, the robot's absolute position (x_a, y_a) in the frame of base station can be calculated by:

$$x_a = d_{rb} \cos(\alpha) \quad [14.9]$$

$$y_a = d_{rb} \sin(\alpha) \quad [14.10]$$

If the distance between the dead reckoned position and the absolute position measured by a scanning laser and a ultrasonic positioning system is less than a selected threshold, the absolute position (x_a, y_a) is fused with the

dead reckoning system by extended Kalman filter. We define $X(k) = (x(k), y(k))$, $X_d(k) = (x_d(k), y_d(k))$ and let $K(k)$ denote the Kalman filter error gain at time k . The extended Kalman filter consists of the following steps:

1) The one-step optimal prediction $\hat{X}(k|k-1)$ should be the dead reckoned position $X_d(k)$ calculated by equations [14.4], [14.5], [14.6] and [14.7]. Thus,

$$\hat{X}(k|k-1) = X_d(k) \quad [14.11]$$

Its propagation error covariance matrix $\tilde{P}(k|k-1)$ can be calculated by:

$$\tilde{P}(k|k-1) = \tilde{P}(k-1|k-1) + Q(k-1) \quad [14.12]$$

where $\tilde{P}(k-1|k-1)$ is the error covariance matrix of optimal estimate $\hat{X}(k-1|k-1)$ at time $k-1$, and $Q(k-1)$ is the diagonal covariance matrix of process noises.

2) Since the robot's position state $X(k)$ can be returned by the scanning laser and ultrasonic positioning system, the state transition matrix $F(k+1, k)$ and observation matrix $C(k)$ should be a unit matrix, so $K(k)$ is calculated by:

$$K(k) = \tilde{P}(k|k-1)[\tilde{P}(k|k-1) + R(k)] \quad [14.13]$$

where $R(k)$ denotes the diagonal covariance matrix of measurement noises.

3) Then, we can get the robot position's optimal estimate $\hat{X}(k|k)$ and update the error covariance matrix $\tilde{P}(k|k)$ using:

$$\hat{X}(k|k) = \hat{X}(k|k-1) + K(k)[X_a(k) - \hat{X}(k|k-1)] \quad [14.14]$$

$$\tilde{P}(k|k) = [I - K(k)]\tilde{P}(k|k-1) \quad [14.15]$$

where $X_a(k)$ denotes the robot absolute position calculated by [14.9] and [14.10] at time k , namely $X_a(k) = (x_a(k), y_a(k))$. After the fusing, we let

$$(x_d(k), y_d(k)) = (x(k), y(k)) \quad [14.16]$$

to eliminate the accumulative error in dead reckoned position.

There is also a scanning laser with a high-resolution encoder mounted on the robot in order to compensate for the accumulative error in robot orientation. The moment the base station is scanned by the robot, it also sends a signal to the robot through RF. The robot then receives the signal and records the robot scanning laser angle β in the robot frame (shown in Figure 14.3). Let α denote the angle the robot's position is at in the base station frame. Therefore,

$$\alpha = \arctan(y(k)/x(k)) \quad [14.17]$$

where $(x(k), y(k))$ is the optimal estimate of the robot's position at time k . So, the robot's heading from the scanning laser can be calculated by:

$$\theta_l(k) = \pi - (\beta(k) - \alpha(k)) + n_l(k) \quad [14.18]$$

where $n_l(k)$ is the measurement noise, which is also assumed to be a zero-mean white Gaussian process with variance σ_l^2 . Figure 14.3 shows the relationship of θ , β and α in [14.18]. Also, if the absolute error between θ and θ_l is less than a selected threshold, for example 3° , the robot's optimal heading can be estimated by:

$$\theta(k) = \frac{\theta_d(k)\sigma_l^2 + \theta_l(k)\sigma_d^2}{\sigma_l^2 + \sigma_d^2}. \quad [14.19]$$

where the dead reckoned robot heading $\theta_d(k)$ is also assumed to be a zero-mean white Gaussian process with variance σ_d^2 . After the estimation, we let $\theta_d(k) = \theta$ to eliminate the accumulative error in dead reckoning angle.

Furthermore, with the scanning laser and the ultrasonic absolute positioning system, the robot can initialize its position and heading at any position and orientation within the system's effective range. To do this, the robot first calculates its initial position through [14.9] and [14.10], and then the heading can be calculated by [14.17] and [14.18].

14.4. Experimental results

We applied the proposed localization system to a lawn robot as shown in Figure 14.4 and our objective is to make it work for a long period of time (e.g. for about one hour) without losing its position, where the lawn robot is designed to mow the lawn with coverage planning rather than randomly.

Therefore, it would not surpass the lawn edge during its mowing work without the assistance of a buried electric wire, which is needed by such current products. In our experiments, the base station is placed at origin (0,0). The scanning laser on the base station and the robot slowly scans from side to side with a period of 5 s.



Figure 14.4. *The experimental lawn robot*

We first conduct an experiment to evaluate the accuracy of the ultrasonic distance measurement on which the performance of the scanning laser and ultrasonic absolute localization system largely depends. The lawn robot is placed at four different distances away from the base station: 240, 1,200, 2,100 and 3,000 cm. When the distance is larger than 35 m, the ultrasonic wave becomes so weak that the receiver array on the robot fails to receive it. Figure 14.5 shows the accuracy. There are 40 measurements at each distance. The respective mean square errors of the four curves in Figure 14.5 are 2.9, 3.0, 5.4 and 8.3 cm. The absolute errors stay well within 10 cm when the distance is approximately 10 m and less. The error becomes larger when the robot gets farther, but stays mostly within 10 cm when the distance is about 20 m. Even at the range of 30 m, the measurement error is acceptable at the 10 cm order of magnitude. The occasional wrong measurements with the absolute error larger than 15 cm can be filtered out by the data fusing strategy.

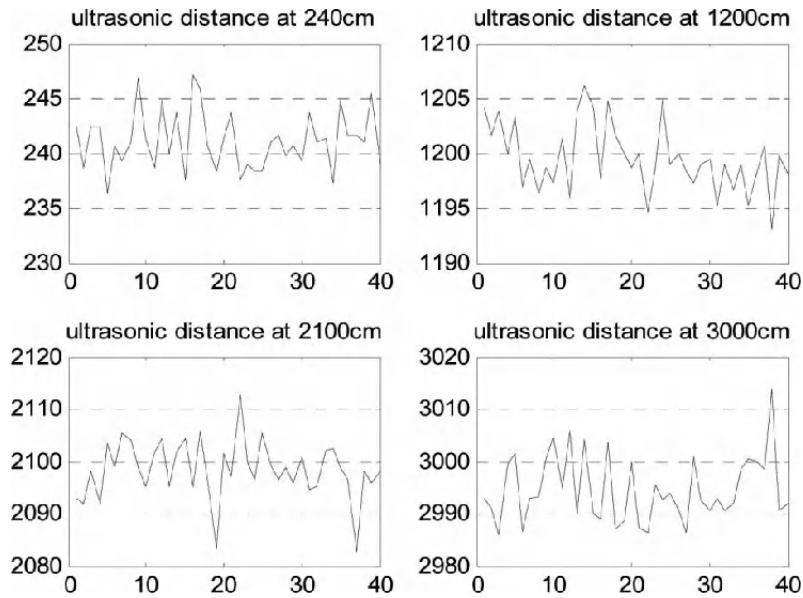


Figure 14.5. Accuracy of scanning laser and ultrasonic positioning system

The next experiment aims at evaluating the accuracy of the scanning laser and ultrasonic absolute positioning system. The robot stops at several specified points in the environment and its position is calculated using the information coming from the scanning laser and the ultrasonic distance measurement. Figure 14.6 shows the results. The crossing point of the dashed line in each subfigure is the robot's specified position, and the circles denote its calculated position by the absolute positioning system, namely (x_a, y_a) . The calculated position in each subfigure seems to be classified into two clusters. This is due to the angle error of the base station scanning laser. Our encoder linked with the scanning laser has a resolution of 1,000 pulses per circle. Therefore, one pulse error in the pulse counter will lead to 0.36° error in the angle measurement. Using encoders of higher resolution will have higher accuracy. The position error led by the angle becomes larger when the robot gets farther away from the base station. Within each point cluster, the error is caused by the fluctuation in ultrasonic distance measurement. As we can see from Figure 14.6, this scanning laser and ultrasonic positioning system is effective when the robot works within 30 m from the base station with the position error being kept at a 10 cm order of magnitude.

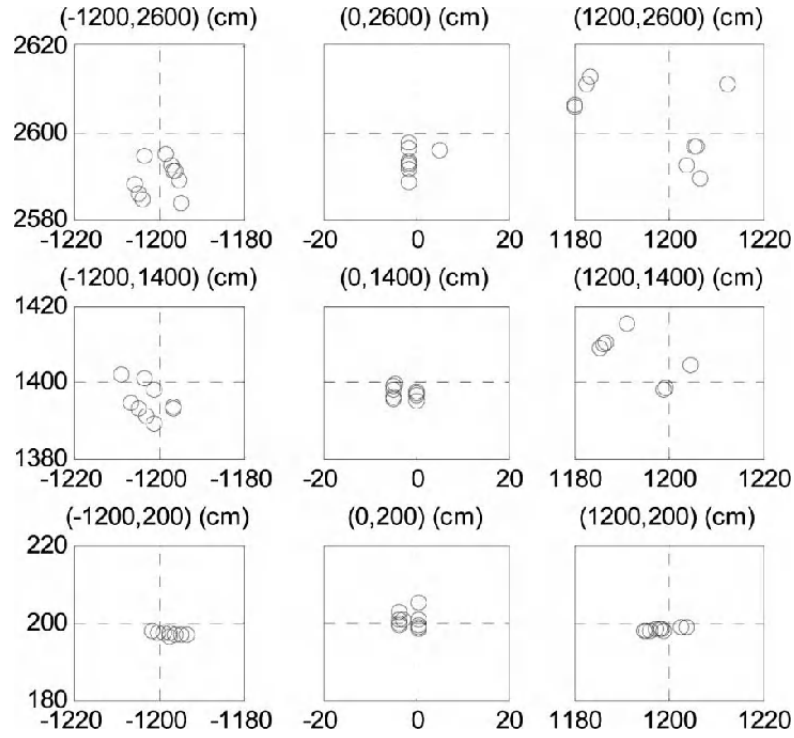


Figure 14.6. Accuracy of ultrasonic distance measurement (cm)

The last experiment evaluates the robot's real-time running performance with the dead reckoning fused with the scanning laser and ultrasonic positioning system. Before the experiment, the parameters of noise models are calculated, where $\sigma_g^2 = 2$, $\sigma_e^2 = 2$, $\sigma_d^2 = 2$, $\sigma_l^2 = 2$, $Q(k) = Q(k-1) = \text{diag}\{1000, 1000\}$ and $R(k) = R(k-1) = \text{diag}\{1000, 1000\}$. The initial conditions of Kalman filter are $\hat{X}(0|0) = (0, 1200)$ and $\tilde{P}(0|0) = \text{diag}\{1000, 1000\}$ (angle unit: degree, distance unit: mm). The experiment was performed on an outdoor lawn. We first manually drive the robot carefully along a $24 \text{ m} \times 24 \text{ m}$ square shape path marked on the lawn by white tape. (The white tape is only used to assist the driver in steering the robot exactly on the desired path, and not used for robot navigation.) In the progress, the robot's calculated position is stored in its memory for every 60 cm traveled.

Then, the stored data are sent to the computer through serial port. Figure 14.7 shows that the calculated position fits well with robot trajectory.

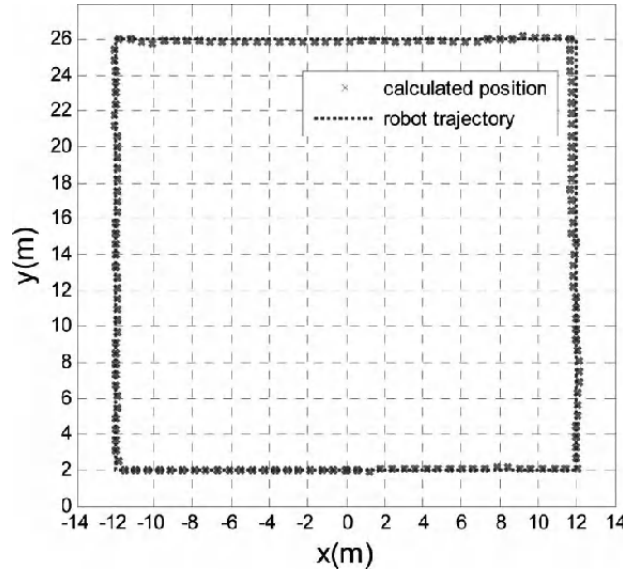


Figure 14.7. Robot trajectory and its calculated position

After that, we let the robot navigate itself automatically along the $24\text{ m} \times 24\text{ m}$ square path, as shown by the dotted path in Figure 14.7, which was also preprogrammed in the robot. Since it is difficult to measure the robot's position error during its navigation, we measure its position when it returns and stops at the starting point $(0, 200\text{ cm})$. This error measuring method was used in [BOR 96] and [CHU 01]. But during our measurement, the robot's software program remains running, and it keeps navigating after the measurement is done. This is in an effort to test the robot's continuous localization ability for long-time running. There are 20 continuous runs with each run lasting about 2.5 min. Both clockwise and anticlockwise measurements are tested. Let (x_r, y_r) denote the tape measured point where the robot returns and (x_c, y_c) denote its calculated position. The error points $(x_r - x_c, y_r - y_c)$ between the returning points and the calculated position are denoted by small squares (clockwise measurements) and circles (anticlockwise measurements) in Figure 14.8. Although the position error becomes reasonably larger when the robot gets farther, the returning point

error at (0, 200 cm) keeps robustly within a 10 cm order of magnitude. The accumulative error of relative dead reckoning is effectively compensated by fusing with the absolute positioning system. This confirms that the proposed method is convergent over long-time running.

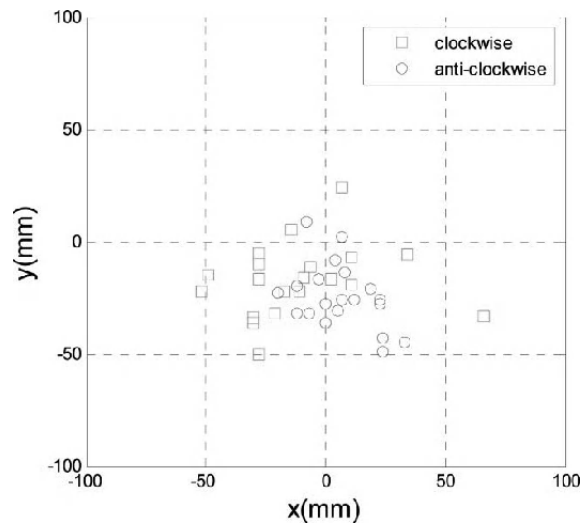


Figure 14.8. Robot trajectory and its calculated position

14.5. Conclusion

This chapter has developed a low-cost, simple and convenient multisensor-based mobile robot localization system. The information from a low-cost gyro and two wheel encoders are fused to get the robot's temporary heading and position. A scanning laser angle measurement and ultrasonic distance measurement system is employed to eliminate the accumulative error in robot's dead reckoned heading and position. The proposed localization system can work both indoors and outdoors and needs no artificial landmarks or accurate map of the robot's working environment, only a base station is needed. Experimental results show that the system works well on a lawn robot. With our localization system, the lawn robot can work on the lawn continuously for a long period of time within 30 m from the base station. The lawn edge can be stored in the robot to avoid it surpassing the edge. Furthermore, the robot can initialize itself at any position within the system's effective range.

14.6. Acknowledgments

This work was supported by the Yangtze Delta Region Institute of Tsinghua University, Zhejiang, China. The first author of this chapter is sincerely appreciative of their kind cooperation. This work was also supported in part by the National Basic Research Program of China (973 Program) under Grant 2012CB821206, by the National Natural Science Foundation of China under Grants 61004021, 61174069 and 61174103, and by the Beijing Natural Science Foundation under Grant 4122037.

14.7. Bibliography

- [BET 97] BETKE M., GURVITS L., “Mobile robot localization using landmarks”, *IEEE Transactions on Robotics and Automation*, vol. 13, no. 2, pp. 251–263, 1997.
- [BOR 96] BORENSTEIN J., FENG L., “Measurement and correction of systematic odometry errors in mobile robot”, *IEEE Transactions on Robotics and Automation*, vol. 12, no. 6, pp. 869–880, 1996.
- [BOR 97] BORENSTEIN J., EVERETT H.R., FENG L., “Mobile robot positioning – sensors and techniques”, *Journal of Robotic Systems*, vol. 14, no. 4, pp. 231–249, 1997.
- [BOR 98] BORENSTEIN J., “Experimental evaluation of a fiber optics gyroscope for improving dead-reckoning accuracy in mobile robots”, *IEEE International Conference on Robotics and Automation*, Leuven, Belgium, pp. 3456–3461, 16–20 May 1998.
- [CHU 01] CHUNG H., OJEDA L., BORENSTEIN J., “Accurate mobile robot dead-reckoning with a precision-calibrated fiber optic gyroscope”, *IEEE Transactions on Robotics and Automation*, vol. 17, no. 1, pp. 80–84, 2001.
- [COH 92] COHEN C., KOSS F.V., “A comprehensive study of three object triangulation”, *SPIE Conference on Mobile Robots*, Boston, MA, USA, pp. 95–106, 18–20 November 1992.
- [FEN 96] FENG L., BORENSTEIN J., “Gyrodometry: a new method for combining data from gyros and odometry in mobile robots”, *IEEE International Conference on Robotics and Automation*, Minneapolis, MN, pp. 423–428, 22–28 April 1996.
- [FON 05] FONT J.M., BATLLE J.A., “Dynamic triangulation for mobile robot localization using an angular state kalman filter”, *European Conference on Mobile Robots*, Ancona, Italy, pp. 20–25, 7–10 September 2005.
- [HAB 07] HABIB M.K., “Real time mapping and dynamic navigation for mobile robots”, *International Journal of Advanced Robotic Systems*, vol. 4, no. 3, pp. 323–338, 2007.

- [HAR 96] HARDT H.J., WOLF D., HUSSON R., "The dead reckoning localization system of the wheeled mobile robot ROMANE", *IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems*, Washington, DC, pp. 603–610, 8–11 December 1996.
- [KOM 94] KOMORIYA K., OYAMA E., "Position estimation of a mobile robot using Optical Fiber Gyroscope (OFG)", *International Conference on Intelligent Robot and System (IROS '94)*, Munich, Germany, pp. 143–149, 12–16 September 1994.
- [ROY 07] ROYER E., LHUILLIER M., DHOME M., "Monocular vision for mobile robot localization and autonomous navigation", *Internal Journal of Computer Vision*, vol. 74, no. 3, pp. 237–260, 2007.
- [SE 05] SE S., LOWE D.G., LITTLE J.J., "Vision-Based global localization and mapping for mobile robots", *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 364–375, 2005.
- [SUZ 92] SUZUKI S., HABIB M.K., YUTA, S., *et al.*, "How to describe the mobile robot's sensor based behavior", *The Journal of Robotics and Autonomous Systems*, vol. 7, nos. 2–3, pp. 227–237, 1992.
- [TSA 98] TSAI C.C., "A localization system of a mobile robot by fusing dead-reckoning and ultrasonic measurements", *IEEE Transactions on Instrumentation and Measurement*, vol. 47, no. 5, pp. 1993–1998, 1998.

Chapter 15

Building of Open-Structure Wheel-Based Mobile Robotic Platform

The chapter describes the development of an open-structure wheel-based mobile robotic platform and a complementary software simulator aimed at research, development and education, as well as the objective benchmarking of this kind of advanced mechatronic system. The main objectives of the work are concerned with solving the problems of development of an autonomous navigation system, motion planning and control of mobile robots in unstructured environments in the presence of mobile and immobile obstacles and system uncertainties such as a variation in tire-ground interaction.

15.1. Introduction

Mobile robots are subjected to many recent research studies [MOR 08, MIN 08] with aim to provide reliable and robust robotic platforms for broad service applications at home, in the office, in public institutions, etc. Mobile robotic platforms make the basis for the building of advanced mechatronic devices such as ambiently intelligent service robots. These systems are expected to be widely utilized in the future in everyday human life, but still many challenges remain to be overcome [HAB 06]. The problems in building such advanced intelligent systems are predominantly related to the success

of solving complex perceptive, cognitive and control tasks related to advance sensing, wireless communication, environmental understanding, trajectory planning as well as adaptation to variable environmental conditions. In that sense, the main tasks to be solved with mobile robots are the development of algorithms for spatial reasoning, autonomous navigation, motion planning and reliable control of system dynamics in conditions of unknown, unpredicted and evolving environments. Therefore, an open-structure mobile robotic platform (a small experimental model, i.e. test-bed station) and corresponding software simulator are needed to support research and education in this field of mechatronics. Accordingly, this chapter deals with the building of such an open-structure test-bed system. Complementary to system development, a hybrid knowledge-based and model-based controller of the robotic system is also designed. The controller developed and integrates navigation and a control module into the unique control architecture designed to enable intelligent system navigation and dynamic control of a wheeled robot in the presence of system uncertainties.

This chapter is organized in the following way. Section 15.2 gives a brief state of the art regarding methods and techniques of advanced navigation and control of autonomous mobile robots. The architecture of the experimental, open-structure wheel-based mobile robotic platform is described in section 15.3. Kinematical and dynamic modeling of the robot system and building of the complementary user-oriented, open-source software simulator dedicated to research and system analysis are presented in section 15.4. Section 15.5 is dedicated to synthesis of advance motion planning and control algorithms. In section 15.6, the simulation and experimental tests are described and the results are analyzed with an aim to validate the proposed algorithms. Section 15.7 gives concluding remarks and some remarks concerning implementation aspects of the open-structure robotic platform. The chapter is finalized by expressing acknowledgments and providing a list of references.

15.2. State of the art

Numerous research studies concerning control of wheeled mobile robots were reported in [MOR 08, MIN 08]. In particular, non-holonomy constraints associated with these systems have motivated the development of highly nonlinear control techniques. For the sake of simplicity, the control methods are developed mainly for unicycle-type and car-like mobile robots. Most of the results can, in fact, be extended to other types of mobile robots (e.g. holonomic), in particular to the systems with trailers. A complementary problem to control the motion of mobile robots is also concerned with global

motion planning and obstacle (mobile and immobile) avoidance in a variety of different static as well as dynamic scenarios (e.g. interaction of several robots). The problem considers sensor-based motion to face the physical issues of real system navigation in a real world. The problem relates to searching for the techniques, how to navigate toward a goal in a confined, troublesome or cluttered environment when the obstacles to avoid are discovered in real time? This is the question that addresses simultaneous motion planning and control as well as obstacle avoidance.

The methods that combine both the global point of view of motion planning and the local point of view of obstacle avoidance have already been developed. How can we consider robot perception at the planning level? This is the so-called sensor-based motion planning. Several variants exist, such as the “bug algorithms” initially introduced in [LUM 87]. However, none of them consider the practical context of non-holonomic mobile robots.

Obstacle avoidance techniques and methods can be categorized in two groups according to [MOR 08]: methods that compute the motion in one step and that do it in more than one step. One-step methods directly reduce the sensor information to a motion control. There are two types of methods:

- The heuristic methods were the first techniques used to generate motion based on sensors. The majority of these works are derived from classic planning methods [LUM 87, CHA 82, STR 84, CHA 85].

- The methods of physical analogies assimilate the obstacle avoidance to a known physical problem. The representatives of these are the potential field methods [KHA 86, KRO 86]. Other works are variants adapted to uncertain models [BOR 89] or that use other analogies [AZA 94, MAS 94, SIN 96].

Methods with more than one step compute some intermediate information, which is processed next to obtain the motion.

- The methods of a subset of controls compute an intermediate set of motion controls, and next choose one of them as a solution. There are two types: (1) methods that compute a subset of motion directions. The vector field histogram [BOR 91] and the obstacle restriction method [MIN 05] can be mentioned as the examples. Another method is presented in [FEI 94]. (2) Methods that compute a subset of velocity controls. The dynamic window approach [FOX 97] and the velocity obstacles [FIO 98] can be taken as examples. Another method based on similar principles but developed independently is the curvature velocity method [SIM 96].

– Finally, there are methods that compute some high-level information as intermediate information, which is translated next in motion. The nearness navigation diagram [MIN 00, MIN 04] is the representative of this method.

Regarding the control techniques applied with mobile robots, there are also several ways applied in engineering practice. Besides Ackermann [WIK], and differential steering, skid steering is the most popular and widely used steering method for wheeled and tracked vehicles. The skid steering mechanism provides high mobility and maneuverability due to strong traction, which makes it suitable for all-terrain traversal. While in differentially steering robots, assumption of no slip can be used to derive adequate models and control laws, this is not the case with skid steering robots, because of higher influence of slip on robot motion. This makes modeling of dynamics and kinematics more difficult, since complex wheels and terrain interaction needs to be accounted in the model, in order to accurately describe the motion of a skid steering robot.

In [CAR 99], a dynamic model for planar motion was developed and it was used to design a feedback linearization based controller for trajectory tracking. A similar dynamic model with the addition of actuator dynamics was introduced in [KOZ 04] and a controller was devised using the back-stepping technique. The property of differential flatness was utilized in [RYU 11] to derive a robust trajectory tracking controller for wheeled mobile robots based on a dynamic and kinematic model with slip. In [YI 07], a pseudo-static friction model is used to capture the interaction between the wheels and ground, and to further develop a dynamic model of a skid-steered wheeled vehicle. Also, an adaptive controller is designed to enable the robot to follow a desired trajectory. The inputs of the dynamic model are the longitudinal slip ratios of the four wheels. Since longitudinal slip ratios are difficult to measure in practice, there is a need for their estimation. A model-based approach for longitudinal slip estimation was introduced in [WAR 08], where a novel tire traction/braking model is presented and used to calculate vehicle dynamic forces in an extended Kalman filter framework.

To summarize the previous consideration, the use of an obstacle avoidance technique with mobile robots in a given scenario is highly dependent on the scenario's nature (static or dynamic, unknown or known, structured or not or its size, for example). Usually, this problem is associated with the integration of motion planning and obstacle avoidance. All the methods outlined in this section have certain advantages as well as disadvantages depending on the navigation context, like uncertain

environments, motion at high speeds and motion in confined or troublesome spaces. Unfortunately, there is no objective metrics available to benchmark the performance of the variety of navigation and control methods in a quantitative way. In that sense, the considerations performed in this chapter represent a contribution to the development of one such mini-benchmarking system, which is open structured, user-oriented and scalable.

15.3. Configuring of the experimental system

The experimental open-structure wheel-based mobile robotic platform is shown in Figure 15.1. It has a modular structure that consists of the following subsystems (modules): (1) four-wheel drive rover, (2) five degrees of freedom (DoF) robot arm for small objects manipulation, (3) IP stereo-vision camera, (4) advanced sensorial system including different localization and navigation sensors (visual and non-visual), (5) PC-based on-board controller and (6) Wi-Fi communication module. A Hagisonic StarGazer optical sensor unit [STA 11] is used as an additional, external module for global localization of the robotic system in the workspace. The high-level system description is presented in Figure 15.2 and the particular modules are described in more detail in the following sections.

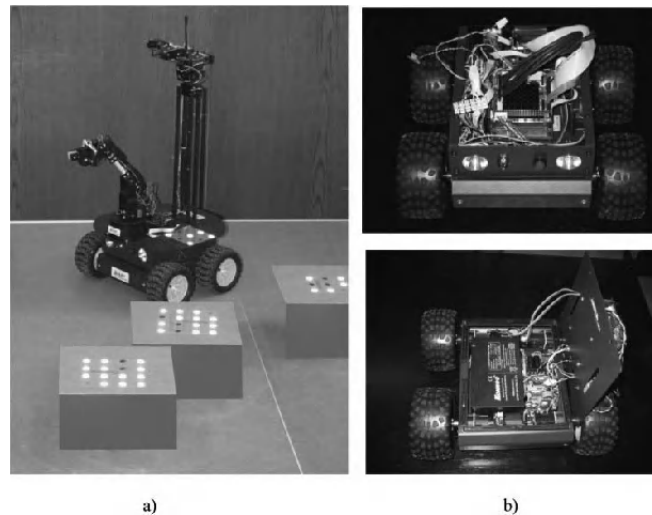


Figure 15.1. a) Assembled 4WD rover; b) Hardware configuration: control and acquisition boards (up) and battery with power-electronics (down). For a color version of this figure please see www.iste.co.uk/mechatroneng.zip

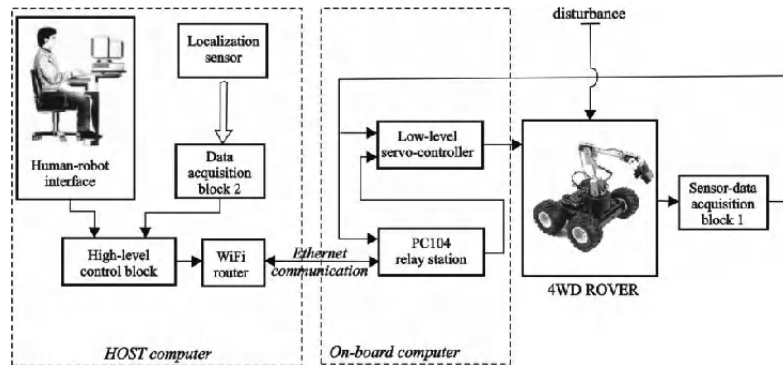


Figure 15.2. High-level system description

15.3.1. Wheel-based mobile robot

The DFRobot 4WD outdoor mobile platform (Figure 15.1) is a small 4WD rover of 6.8 kg payload designed for the research purposes [DFR 11]. The robot is supported by four wheels that are driven in a differential configuration. The wheels are powered by geared direct current (DC) motors with mounted incremental encoders. The sensory system of the mobile platform consists of infrared (IR) and ultrasound proximity sensors as well as an three-axis gyroscope and an three-axis accelerometer. A five-DoF robot arm AL5A [LYN 11] is mounted on the mobile platform. It is powered by 5 RC servo-motors. The control system of the mobile platform (Figure 15.3) is based on the master PC-104 board [ADL 11] and a slave board for control of the RC servo/motors. A PC-104 board is powered by an Intel® Atom™ processor 512 MB RAM, and 2 GB SSD flash drive master board also contains a bundle of interfaces: USB, RS232 and connectors to various peripheral boards. In addition, a USB-to-Wi-Fi adaptor is used for wireless communication with main computer.

15.3.2. System for localization and obstacle detection

StarGazer [STA 11] is chosen as an indoor localization system appropriate for this project. An on-board DSP processes an IR image, which is reflected from passive landmarks on the scene. Each landmark has a unique ID. Outputs from the processing unit are the position and the heading angle of the landmark. The system is rather accurate (error less than 1 cm), reliable and resistant to fluorescent light and sunshine. Stargazer is connected to the PC server via RS232 bus.

The basic tasks of navigation and control of mobile robotic systems are tasks of obstacle and collision avoidance. For that purpose, with the 4WD rover (Figure 15.1), two kinds of sensors are implemented: (1) Mindsensors [MIN 11] – a high-precision medium range (1.5 m) IR distance sensor, and (2) Devantech [DEV 11] – a SRF10 ultrasonic scanning range finder for detection proximity of obstacles and mapping environment. The robot is equipped with a wireless IP stereo-vision system [SUR 11].

15.3.3. Architecture of the on-board controller

The control architecture is shown in Figure 15.3. The mobile robot controller consists of the following elements: a ADLS15PC – Intel® Atom™ CPU 1.60 GHz PC104 processor board, a PC104 acquisition board, a ENCDA PC104 analog encoder board, a DIGIO PC104 universal board, a DC–DC converter block that provides power supply for the system, a motor driver, a robotic arm and a stereo camera controller, a network of ultrasonic and IR sensors, integrated three-axis accelerometer and three-axis gyroscope.

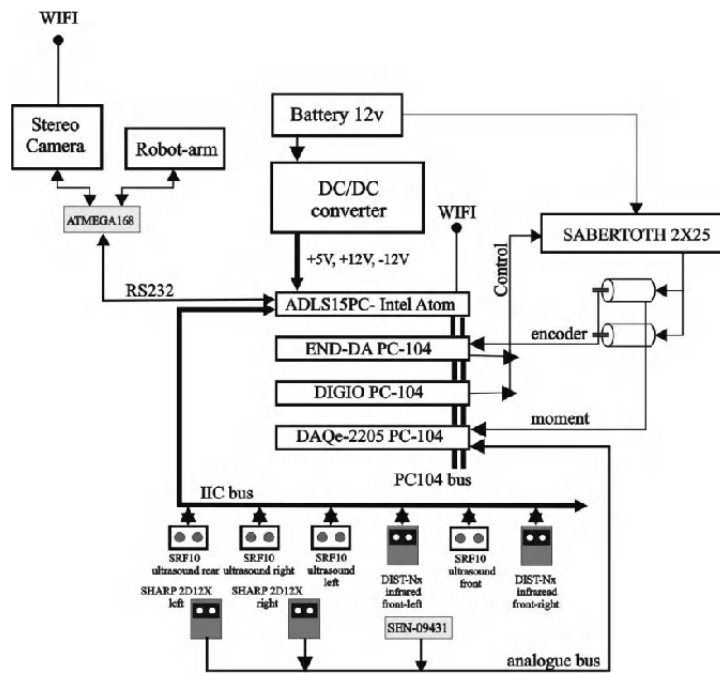


Figure 15.3. Architecture of the system on-board controller

The PC104 controller has the following characteristics: Intel® Atom™, dual core processor, 1.60 GHz operating frequency, Intel SCH US15W Chipset, 1 GB DDR2-DRAM, CRT/LVDS Interface Onboard, 10/10/1000 Base-T Ethernet, 8 × USB2.0 ports, PATA Storage Interface, COM1, COM2, LPT1, SM Bus Interface, RTC and Watchdog Timer, High-Definition Audio, 4 GB On-board SSD. A UBUNTU operating system has been installed to perform monitoring and control of the whole system, to receive commands from a base station via Wi-Fi network paths and to execute received commands. An encoder and digital to analog output (ENCDA) universal board has been developed specifically to provide motor control and acceptance of signals from the encoder. A digital input-output (DIGIO) PC104 universal board was designed to allow reading of 16 digital optically isolated general purpose signals and activation of eight digital outputs that might be configured differently (PNP type transistor, NPN type transistor, NMos and Relay). In the robot controller, the DIGIO board was used for control of the lights on the robot and control of the operating voltage for the motor driver and robotic arm driver. The SABERTOOTH 2 × 25 driver is used as the motor driver since it provides independent control of two motors simultaneously. As a control signal for the driver, voltage signal from −10 to 10 V is used, by which the driver generates the appropriate output (voltage and current) to the motor joint.

A network of embedded IR and ultrasonic obstacle detectors provides simultaneous and independent reading and detection of the obstacles within a 360° circle. Obstacle sensors are implemented in a network through a standard IIC bus implemented on the LPT1 port, as well as through direct reading of the sensor connected to acquisition card. Acquisition card DAQe-2205 features are as follows: Analog 64 single-ended inputs or 32 differential inputs; resolution/sample rate 16-bit at 500 kS/s; input ranges: unipolar 0 to +1.25, +2.5, +5 and +10 V; bipolar ±1.25, ±2.5, ±5 and ±10 V; A/D FIFO size 1,024 samples; trigger modes pre-/post-/middle-/delay-/repeat-trigger; analog outputs: two 12-bit waveform generation channels; 1 MS/s update rate; output ranges: 0 to +10, ±10 V, 0 to AOEXTREF, ±AOEXTREF; D/A FIFO size 1,024 samples; digital I/O 24 lines (82C55) all boards; counter/timer: two 16-bit counter/timers. With a PC104 acquisition card it is possible to measure and monitor the current of the vehicle motors and thus perform more advanced control techniques of the vehicle based on the position as well as on the force in the wheels. This acquisition card is connected to the following sensors: the three-axis accelerometer and the three-axis gyroscope SEN-09431 and two side IR sensors SHARP 2D12X. An inertial sensor has the following characteristics: pitch, yaw and roll gyro outputs, 1 × and 4 × amplified (0.83 and 3.33 mV/°/s sensitivity,

respectively) $\pm 300^\circ/\text{s}$ range, x -, y - and z -axis accelerometer outputs, 300 mV/g sensitivity, ± 3 g range; all necessary filtering components, access to gyro's self-test, power down and high-pass filter reset pins. The IR obstacle sensor use the IR signal to measure object distance from 10 to 80 cm with analog output. It has been mounted on the sides of the vehicle to detect obstacles in the space on the left and right. The sensors network on the IIC bus consists of four SRF10 ultrasonic obstacle detectors mounted on all four sides of the vehicle, and two IR obstacle sensors DIST Ns medium-V2.

Maximum range of the SRF10 is set by an internal timer. By default, this is 65 ms or the equivalent of 11 m in range. This is much further than 6 m for which SRF10 is actually capable. It is possible to reduce the time that the SRF10 takes to detect an echo, and hence the range, by writing to the range register at location 2. The range register can be set in steps of about 43 mm (0.043 m or 1.68 in) up to 11 m. The range is $((\text{range register} \times 43 \text{ mm}) + 43 \text{ mm})$, so setting the range register to 0 (0×00) gives a maximum range of 43 mm. Setting the range register to 1 (0×01) gives a maximum range of 86 mm. More usefully, 24 (0×18) gives a range of 1 m and 93 ($0 \times 5D$) gives of 4 m. Setting 255 ($0 \times FF$) gives the original 11 m ($255 \times 43 + 43$ is 11,008 mm). An IR obstacle sensor DIST-Nx is an optical distance sensor (various SHARP sensors) with digital (IIC) interface. This can be used to measure precise distances from an obstacle. The distance range of sensor variants is medium range: 10–80 cm (with the highest precision in a zone ranging from 10 to 40 cm). The sensor detects distance based on the angle of arriving reflected IR. The sensor is mounted on the front angles of the vehicle.

A controller using RS232 interface controls the robotic arm that is mounted on the front of the vehicle. The robotic arm has five DoF as well as a service grab. The RC servo motors were used in the arm. The microcontroller, ATMEL ATMEGA168-20PU, directly controls RC servo motors by generating a pulse width modulation (PWM) signal with a period of 20 ms for each DoF. This microcontroller controls a pan-tilt stereo camera module Surveyor Stereo Vision System (SVS) SRV-1 Blackfin [SUR 11], as well. This is achieved through a serial RS232 link, with the band rate of 19,200. The schematic representation of the controller is given in Figure 15.2.

The complete system consists of two mutually connected modules communicated by Wi-Fi interface. A 4WD rover with on-board control is subordinated to the high-level control module installed at the host computer (basis station) as shown in Figure 15.3. A corresponding graphical user interface (GUI) ensures remote monitoring, that is supervisory control and data acquisition.

15.4. Modeling and simulation of the system

For designing of control algorithms and their validation, prior to an experimental testing, appropriate modeling of the wheeled robot (Figure 15.1) with differential (skid) steering capabilities is done. A corresponding software simulator is built based on the model developed. A nonlinear model of the 4WD rover is assumed as an appropriate representative that takes into account the robot capability to move on a sloped surface, too. The appropriate model that fits the system's physical capabilities in a satisfactory accurate way is presented in Figures 15.4 and 15.5. In a general case, it is assumed that the surface slope appears in both longitudinal γ_x as well as lateral γ_y direction of motion as shown in Figure 15.5. A complementary kinematical model of rover tires is presented in Figure 15.4, too. Special attention in this chapter is dedicated to modeling of nonlinear tire dynamics. The importance of parameter estimation of tire-ground adhesion will be emphasized, too. Both the extremely nonlinear nature of rubber tires and the corresponding parametric uncertainties regarding tire rolling and slipping during skid steering maneuvers are considered in this chapter. In the following sections, appropriate modeling of 4WD robot kinematics and dynamics is presented.

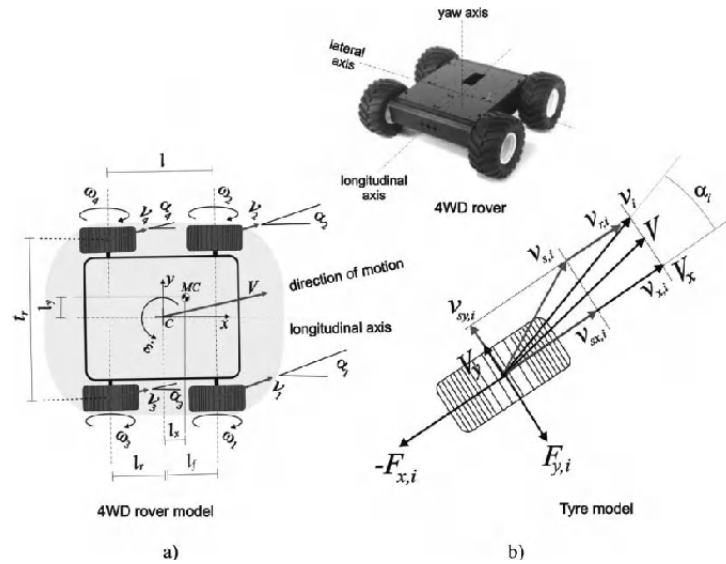


Figure 15.4. a) Kinematical model of 4WD rover considered in this chapter and b) corresponding tire kinematical model

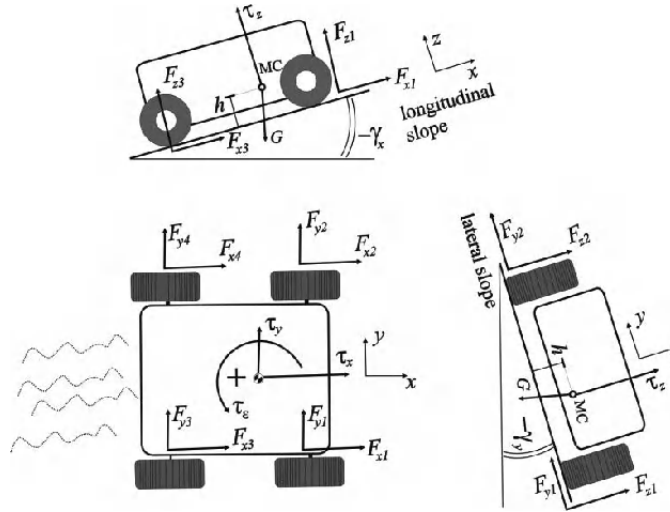


Figure 15.5. 4WD rover dynamic model – tire forces

15.4.1. Kinematical model

Holonomic mechanisms are able to perform controlled movements through every available DoF within its particular task space. The non-holonomic mechanisms are capable of performing fewer controlled DoF than are actually available within its task space. A robot capable of differential (skid) steering, such as the rover presented in Figure 15.1, is able to move from one point to a different point and orientate within the same two-dimensional (2D) task space. However, it is only able to perform a single displacement (forward/backward) and a single rotation (turn, pivot or spin). Thus, it can execute only two controlled DoF, which is one less than the available (forward/backward, right/left and turn) in its task space. A kinematical model of the 4WD rover can be defined by the vector relation:

$$\begin{bmatrix} V \\ \dot{\epsilon} \end{bmatrix} = \begin{bmatrix} r_t/2 & r_t/2 & r_t/2 & r_t/2 \\ \frac{r_t \cos(\zeta)}{d} & -\frac{r_t \cos(\zeta)}{d} & \frac{r_t \cos(\zeta)}{d} & -\frac{r_t \cos(\zeta)}{d} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} \quad [15.1]$$

where V and $\dot{\epsilon}$ represents the forward (lump) speed and the robot yaw-rate, respectively; r_t is a tire radius, $\omega_i, i=1, \dots, 4$ are particular tire angular

velocities; d is a geometry parameter, that is the corresponding planar distance of the robot tire centers from the robot geometry center C . The parameters used in the model presented in Figure 15.4 have the following meaning: t_r represents a track of the rover, l is the longitudinal distance between the front and rear wheel axes, l_x and l_y are corresponding distances of the body mass center (MC) from the rover geometry center C (point of crossing of the longitudinal and lateral axes) and h is the distance of the MC with respect to the ground surface (Figure 15.5).

Parameter d is calculated from the relation $d = 1/2 \cdot \sqrt{l^2 + t_r^2}$; angle ζ is the corresponding design-related parameter calculated from $\text{tg}(\zeta) = l / t_r$. The forward speed V has two components (Figure 15.4): the longitudinal $V_x \equiv \dot{x}$ and the lateral $V_y \equiv \dot{y}$. The speed is calculated from the equation:

$$V = \sqrt{\dot{x}^2 + \dot{y}^2} \quad [15.2]$$

The tire kinematical model takes into account both – a tire rotation as well as a tire slipping on the ground surface. Consequently, the velocity vector of the i th tire v_i consists of two components: the slipping velocity $v_{s,i}$ and the speed $v_{r,i}$ due to a tire rotation (Figure 15.4). The slipping velocity $v_{s,i}$ depends not only on the tire revolution per minute (rpm) but also on the parameters of tire–ground interaction. The kinematical indices that represent a measure of tire slipping on the support surface in the longitudinal and lateral directions are known [ROD 02, PAC 93] as the tire slip ratio s_i and the tire slip angle α_i (Figure 15.4). The tire slip ratio and tire slip angle can be determined using the following kinematical relations [ROD 02, PAC 93]:

$$s_i = -\frac{v_{sx,i}}{v_{x,i}} = -\frac{v_{x,i} - v_{r,i}}{v_{x,i}} = -\frac{v_i \cos(\alpha_i) - r_i \omega_i}{v_i \cos(\alpha_i)}, \quad i = 1, \dots, 4 \quad [15.3]$$

$$\text{tg } \alpha_i = -\frac{v_{sy,i}}{v_{x,i}}, \quad i = 1, \dots, 4 \quad [15.4]$$

The tire speed v_i (Figure 15.4) represents speed of motion of the i th tire MC. The tire slip angle α_i represents an angle defined with respect to the

longitudinal direction of motion (Figure 15.4). Only in an ideal case: the slipping velocity $v_{s,i}$ can be neglected and then it is assumed that the tires move by the speeds caused only by the rotation. The ideal case, for example, corresponds to the case of motion at low speed over a high frictional surface. The linear (translational) tire speeds $v_i, i=1, \dots, 4$ are calculated from the following kinematical relations assuming the model presented in Figure 15.4:

$$\begin{aligned} v_1 &= \sqrt{(\dot{y} + l_f \dot{\epsilon})^2 + (\dot{x} - t_r / 2 \dot{\epsilon})^2} \\ v_2 &= \sqrt{(\dot{y} + l_f \dot{\epsilon})^2 + (\dot{x} + t_r / 2 \dot{\epsilon})^2} \\ v_3 &= \sqrt{(\dot{y} - l_r \dot{\epsilon})^2 + (\dot{x} - t_r / 2 \dot{\epsilon})^2} \\ v_4 &= \sqrt{(\dot{y} - l_r \dot{\epsilon})^2 + (\dot{x} + t_r / 2 \dot{\epsilon})^2} \end{aligned} \quad [15.5]$$

where l_f and l_r represent corresponding geometry parameters shown in Figure 15.4. According to Figure 15.4, these lengths are $l_f = l_r = (1/2) \cdot l$. The directions of the particular tire speeds calculated in equation [15.6] are defined by the corresponding tire slip angles α_i calculated by use of the expressions:

$$\begin{aligned} \operatorname{tg}(\alpha_1) &= \frac{\dot{y} + l_f \dot{\epsilon}}{\dot{x} - t_r / 2 \dot{\epsilon}}, & \operatorname{tg}(\alpha_2) &= \frac{\dot{y} + l_f \dot{\epsilon}}{\dot{x} + t_r / 2 \dot{\epsilon}} \\ \operatorname{tg}(\alpha_3) &= \frac{\dot{y} - l_r \dot{\epsilon}}{\dot{x} - t_r / 2 \dot{\epsilon}}, & \operatorname{tg}(\alpha_4) &= \frac{\dot{y} - l_r \dot{\epsilon}}{\dot{x} + t_r / 2 \dot{\epsilon}} \end{aligned} \quad [15.6]$$

15.4.2. Model of robot dynamics

The state vector of the considered robotic system (Figure 15.4) and its derivative are defined by the following relations:

$$\begin{aligned} q &= [x \ y \ z \ \epsilon]^T \\ \dot{q} &= [\dot{x} \ \dot{y} \ \dot{z} \ \dot{\epsilon}]^T \end{aligned} \quad [15.7]$$

where x , y and z represent the coordinates of the robot MC defined with respect to the inertial coordinate system attached to the ground surface (see

Figure 15.4). Angle ε represents the corresponding yaw angle of the robot body about the z-axis. The robot dynamic model is defined in the state space in a way described in the following section.

15.4.2.1. Robot rigid-body dynamics

A 3D-model of the chosen robotic system (Figure 15.1) is presented in Figure 15.5. The model assumed describes robot dynamics with a satisfactory accuracy taking into account the most significant dynamic effects of the system. The model is presented in its general vector form:

$$T = H(q) \cdot \ddot{q} + h_{cgg}(q, \dot{q}) - F_w(\dot{q}) \quad [15.8]$$

where $T \in \mathfrak{R}^{4 \times 1}$ represents vector of the generalized forces and torques acting in the body MC. The generalized vector T has four components (Figure 15.5). T_x , T_y and T_z are generalized forces acting in the MC in longitudinal x , lateral y and z direction perpendicular to the ground support. The generalized torque T_ε produces spinning of the robot body about the z -axis; $H \in \mathfrak{R}^{4 \times 4}$ is the inertia matrix; $h_{cgg} \in \mathfrak{R}^{4 \times 1}$ is the vector of centrifugal, Coriolis and gravity forces and torque; $F_w \in \mathfrak{R}^{4 \times 1}$ is the vector that includes the main external resistance forces (torque) such as: aerodynamic resistance, tire rolling resistances and Coulomb friction during a robotic motion. In order to achieve desired motion q and $\dot{q}$ defined in equation [15.7], the vector of the generalized forces and torque T , defined in [15.8], has to generate motion defined by the forward speed V and direction ε (robot cornering). In order to achieve desired forward speed and direction of motion, the robot has to produce corresponding tire forces and yaw torque τ in such a way to achieve that $T = \tau$. A vector that takes into account the tire forces (traction/braking and payload) and corresponding yaw moment that produces rover spinning about its z -axis is defined by:

$$\tau = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \\ \tau_\varepsilon \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^4 F_{xi} \\ \sum_{i=1}^4 F_{yi} \\ \sum_{i=1}^4 F_{zi} \\ M_z \end{bmatrix} \quad [15.9]$$

$$M_z = (F_{x1} + F_{x3}) \cdot \left(\frac{t_r}{2} + l_y \right) - (F_{x2} + F_{x4}) \cdot \left(\frac{t_r}{2} - l_y \right) \\ + (F_{y1} + F_{y2}) \cdot (l_f - l_x) - (F_{y3} + F_{y4}) \cdot (l_r + l_x), \quad [15.10]$$

where F_{x_i} , F_{y_i} ($i=1, \dots, 4$) are corresponding longitudinal and lateral tire forces due to traction or braking (Figure 15.5). Tire load F_{z_i} distribution depends on the distribution of robot masses as well as on the robot's position on a sloped surface (Figure 15.5).

The corresponding matrix and vectors of the model [15.8] are [ROD 02]:

$$H = \begin{bmatrix} m & 0 & 0 & 0 \\ 0 & m & 0 & 0 \\ 0 & 0 & m & 0 \\ 0 & 0 & 0 & I_z \end{bmatrix} \quad [15.11]$$

$$h_{cog} = \begin{bmatrix} -m \dot{y} \dot{\epsilon} + \dot{z} \dot{\gamma}_x + m g \sin(\gamma_x) \\ m \dot{x} \dot{\epsilon} - \dot{z} \dot{\gamma}_y - m g \sin(\gamma_y) \cos(\gamma_x) \\ -\dot{x} \dot{\gamma}_x + \dot{y} \dot{\gamma}_y + m g \cos(\gamma_y) \cos(\gamma_x) \\ 0 \end{bmatrix} \quad [15.12]$$

where m is the lump mass of the robotic system, I_z is the robot's axial moment of inertia with respect to the z -axis and g is acceleration due to gravity. The total (resultant) resistance vector F_w including aerodynamic resistance, tire rolling resistance and Coulomb friction, is given below [ROD 02]:

$$F_w = \begin{bmatrix} -K_x \dot{x}^2 - \sum_{i=1}^4 f_{r_i} F_{z_i} \cos \zeta_i \\ -K_y \dot{y}^2 - \sum_{i=1}^4 f_{r_i} F_{z_i} \sin \zeta_i \\ 0 \\ \sum_{i=1}^4 M_{\alpha i} \end{bmatrix}, \quad [15.13]$$

where K_x , K_y represents corresponding air resistance coefficients of the robot's body; $\sum_{i=1}^4 M_{\alpha i}$ is sum of the tire self-aligning torques (tire resistance to

spinning about the z -axis) due to rover cornering; f_{r_i} is a rolling resistance coefficient of the particular i th tire and F_{z_i} represents corresponding tire payload depending on position of the MC (i.e. robot mass distribution) and the slope of a ground surface.

15.4.2.2. Nonlinear tire model

A considerable number of different models of tire force and moment-generating properties have been proposed in the available literature. The standard (proposed by the Society of Automotive Engineers (SAE)) in description of vehicle tire dynamics represents the so-called magic formula tire model originally introduced by Pacejka and Bakker [PAC 93]. The model provides a set of mathematical formulas from which the forces and moment acting from road to tire can be calculated at longitudinal and lateral slip conditions, which may occur simultaneously. The model aims at an accurate description of measured steady-state tire behavior. The coefficients of basic formula that are given in this section represent typifying quantities of the tire characteristic. By selecting proper values, the characteristics for either side force, aligning torque or fore and aft force can be obtained. The core of the nonlinear model is formed by the formula that has become known under the name “magic formula”. This formula is capable of producing force and moment characteristics at pure slip conditions, that is either pure cornering or pure braking or driving. The formula (model) expresses the side force F_y , the aligning torque M_α and the longitudinal force F_x as a function of two arguments – the side slip angle defined by [15.4] and the longitudinal tire slip ratio determined by [15.3], respectively. The general form of the formula, which holds for a given value of vertical tire load, is:

$$f(u) = D \cdot \sin \left\{ C \cdot \arctg \left[B \cdot u - E \cdot (B \cdot u - \arctg(B \cdot u)) \right] \right\} \quad [15.14]$$

The empiric nonlinear Pacejka’s tire model is shown in Figure 15.6. The magic formula $f(u)$ typically produces a curve that passes through the origin $u = f = 0$, reaches a maximum and subsequently tends to a horizontal asymptote f_s (Figure 15.3). For constant coefficients B , C , D and E , the curve shows an antisymmetric shape with respect to the origin.

The formula is capable of producing characteristics that closely match measured curves for longitudinal F_x force, side (lateral) F_y force and self-aligning torque M_α as functions of their respective slip quantities: the slip

angle α [15.4] and longitudinal slip ratio s [15.3]. The output variable stands for either F_x , F_y or M_α and the input u may represent either s or α .

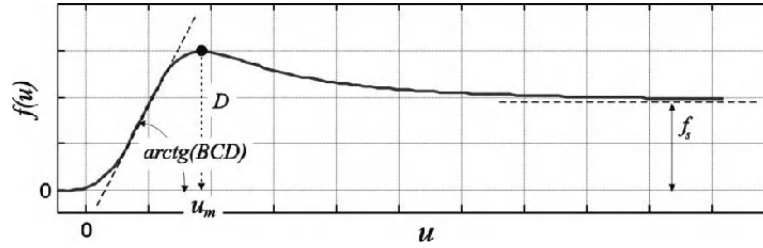


Figure 15.6. General shape of the “magic formula” tire model to be used to describe tire nonlinear dynamics of the skid-steered 4WD rover (Figure 15.1)

Figure 15.6 illustrates the meaning of some of the coefficients with the help of typical side force or longitudinal force characteristics [PAC 91, PAC 93, PAS 94]. Coefficient D (Figure 15.6) represents the peak value (with respect to the abscissa) and the product BCD corresponds to the slope at the origin $u = f = 0$. The factor C controls the limits of the range of the argument of the sine function. It thereby determines the shape of the resulting curve. Typical empirical values of the shape factor C are 1.30 for the side force F_y , 2.40 for the self-aligning torque M_α and 1.65 for the longitudinal (traction/brake) force F_x characteristic. The factor B is left to control the slope at the origin and is, therefore, called the stiffness factor. The remaining factor, E , appears to be necessary to influence the curvature near the peak of the curve. At the same time, E controls the slip u_m (Figure 15.6) at which the peak occurs (if present).

$$E = \frac{B \cdot u_m - \operatorname{tg}\left(\frac{\pi}{2 \cdot C}\right)}{B \cdot u_m - \operatorname{arctg}(B \cdot u_m)} \quad [15.15]$$

The asymptotic value that $f(u)$ approaches at large slip values equals:

$$f_s = D \cdot \sin\left(\frac{\pi}{2} \cdot C\right) \quad [15.16]$$

From these expressions, the initial values for C and E may be obtained for the least square regression procedure to acquire an optimal match to

measured data. More often, C is given a fixed value, typical for the kind of curve. A tire payload, that is tire torque that produces its rotation is calculated from:

$$\tau_t = F_x \cdot r_t \quad [15.17]$$

where r_t is the tire radius. Tire payload τ_t as well as tire angular velocity (tire rpm) ω are used as feedback values for the servo-control of 4WD robot rover [SME 99, RAJ 78].

15.4.2.3. Model of actuators

DC motors are widely used as reliable actuators of mobile robots (Figure 15.1). According to [SME 99], the third-order mathematical model of a DC actuator can be written in the form:

$$\dot{x}_i = C_i x_i + f_i \hat{\tau}_i + d_i N(u_i), \quad i = 1, \dots, 4 \quad [15.18]$$

where a 3D state vector for such system can be represented by:

$$x_i = [q_i \quad \dot{q}_i \quad i_{r_i}]^T, \quad i = 1, \dots, 4 \quad [15.19]$$

The variables used in relations [15.18] and [15.19] have the following meaning: q_i and $\dot{q}_i$ are rotor position (angle) and angular velocity of the i th particular wheel actuator; i_{r_i} represents corresponding rotor current; $\hat{\tau}_i$ is the motor torque and $N(u_i)$ is the control voltage of a saturation type as defined in the following relation:

$$N(u_i) = \begin{cases} -u_{i,\max} & \text{for } u_i \leq -u_{i,\max} \\ u_i, & \text{for } -u_{i,\max} < u_i < u_{i,\max} \\ u_{i,\max} & \text{for } u_i \geq u_{i,\max} \end{cases} \quad [15.20]$$

According to [SME 99], the system matrices are:

$$C_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -B_c/J_r & C_M/J_r \\ 0 & -C_E/L_r & -R_r/L_r \end{bmatrix}; \quad f_i = \begin{bmatrix} 0 \\ -1/J_r \\ 0 \end{bmatrix}; \quad d_i = \begin{bmatrix} 0 \\ 0 \\ 1/L_r \end{bmatrix} \quad [15.21]$$

Where C_M and C_E are the moment constant and the constant of electromotor force, L_r and R_r are the rotor inductivity and resistance, J_r is

the rotor moment of inertia, and B_c is the viscous friction coefficient reduced to the output shaft. Index “ i ” is omitted due to the simplification reasons.

15.5. Motion planning and control

The motion planning problem for non-holonomic systems, such as considered skid-steered 4WD rover (Figure 15.1), can be stated as follows: given a goal point in the environment with obstacles (mobile and immobile) in the workspace whose relative position (distance and azimuth angle) with respect to the robot is measurable, a robot subject to non-holonomic constraints, an initial configuration and a goal configuration and an admissible collision-free path between the initial and goal configurations are searched. Solving this problem requires us to take into account both the configuration space constraints due to obstacles and the non-holonomic constraints of the robotic system. The tools developed to address this issue thus combine motion planning and control theory techniques. In the following section, more details are given regarding the solution of these problems.

15.5.1. Motion planning

The objective of motion planning techniques is to compute a collision-free trajectory to the target configuration that complies with the vehicle constraints. This assumes a perfect model of the robot and scenario in advance. The advantage of these techniques is that they provide complete and global solutions of the problem. Nevertheless, when the surroundings are unknown and unpredictable, these techniques fail. One approach is to the map environment and to generate trajectories in real time [HAB 07], although these techniques are usually computationally expensive. A complementary way to address the motion problem in an unknown environment is obstacle avoidance. The objective is to move the vehicle toward a target location free of collisions with the obstacles detected by the sensors during motion execution. The advantage of reactive obstacle avoidance is to compute a motion by introducing the sensor information within the control loop, used to adapt the motion to any contingency incompatible with initial plans. The main cost of considering the reality of the world during operation is locality. In this instance, if global reasoning is required, a trap situation could occur. Despite this limitation, obstacle avoidance techniques are mandatory to deal with mobility problems in unknown and evolving surroundings.

The approach considered in this study assumes that the robot is equipped with corresponding proximity and range-finder sensors for detection of obstacles (as described in section 15.3.1) in surrounding, and that it operates in an informatically structured environment. This means that the robot is enabled to acquire satisfactory accurate information about its global position in the workspace and to determine a precise position of the robot target point in any time.

For accurate navigation and motion planning in the presence of obstacles, a robot needs to sense the following variables: (1) distance to surrounding obstacle(s), (2) its relative position with respect to obstacles (an angle calculated with respect to the robot longitudinal axis of symmetry), (3) relative position (azimuth angle) of the robot with respect to the target point and (4) in case of collision avoidance of mobile obstacle(s), a robot requires additional information regarding the obstacles speed of motion and direction in order to achieve a collision-free path. For motion planning with mobile robots, the fuzzy inference system (FIS) is an appropriate technique that enables accomplishing of such cognitive tasks. In this study, an FIS with two input and two output variables, similar to FIS in [MIT 10], is designed as the high-level motion planning module. The corresponding membership functions of the input variables are presented in Figure 15.7.

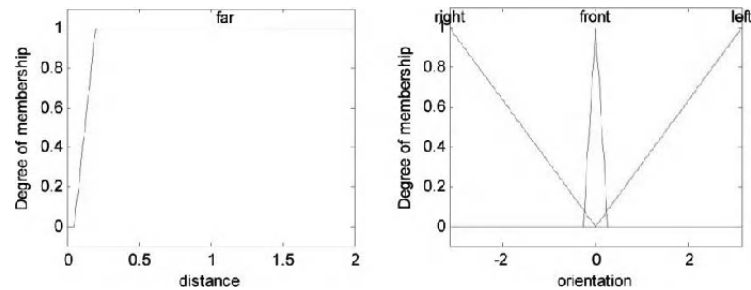


Figure 15.7. Fuzzy membership functions of input variables “distance” and “orientation”

The input variable “distance” possesses only one membership function “far”, which is of a trapezoidal type with parameters $[0.05 \ 0.1 \ 2 \ 2]$ (Figure 15.7). The input variable “orientation” is composed out of three membership functions: “right”, “front” and “left” that are of a triangular type. Its parameters are $[-\pi \ -\pi \ 0]$, $[-\pi/10 \ 0 \ \pi/10]$ and $[0 \ \pi \ \pi]$, respectively (Figure 15.7).

The output membership functions of zero-order Sugeno-type fuzzy system are singletons (Table 15.1). For the output “velocity”, fuzzy singletons (constants) are $V_{\max}$, V_{turn} and 0. For the output “angular velocity”, constants are $-\omega$, 0 and ω .

	Velocity			Angular velocity		
	$V_{\max}$	V_{turn}	0	ω	0	$-\omega$
value	0.5 m/s	0.2 m/s	0 m/s	2 rad/s	0 rad/s	-2 rad/s

Table 15.1. Fuzzy singleton values

The assumed rules of the FIS base are as follows:

- 1) If “distance” is “far”, and “orientation” is “right”, then “velocity” is V_{turn} and “angular velocity” is ω .
- 2) If “distance” is “far”, and “orientation” is “front”, then “velocity” is $V_{\max}$ and “angular velocity” is 0.
- 3) If “distance” is “far”, and “orientation” is “left”, then “velocity” is V_{turn} and “angular velocity” is $-\omega$.
- 4) If “distance” is “not far”, then “velocity” is 0 and “angular velocity” is 0.

The membership functions and the fuzzy rules are designed by using the Matlab®/Simulink Fuzzy Logic Toolbox [MAT 11].

Taking into account the generated output variables of the fuzzy path planner as well as bearing in mind the state variables of the system q and $\dot{q}$ defined by [15.7], a referent (desired) motion is possible to be defined. If the referent forward speed V_0 and the referent angular velocity (yaw rate) $\dot{\epsilon}_0$ of the 4WD rover are assumed to be obtained from the fuzzy motion planning block, then the referent state variables q_0 , $\dot{q}_0$ and $\ddot{q}_0$ of the system can be established in the following way [ROD 10b]:

$$\begin{aligned}
 \dot{x}_0 &= V_0 \cos(\epsilon_0) \\
 \dot{y}_0 &= V_0 \sin(\epsilon_0) \\
 \dot{\epsilon}_0 &= \dot{\epsilon}_0
 \end{aligned}
 \tag{15.22}$$

where ε_0 is the referent, that is nominal yaw angle determined previously from the expression:

$$\varepsilon_0 = \int \dot{\varepsilon}_0 \cdot dt, \quad t \in (0, T] \quad [15.23]$$

The vector of nominal speed values $\dot{q}_0$ of the robotic system has the form:

$$\dot{q}_0 = [\dot{x}_0 \quad \dot{y}_0 \quad 0 \quad \dot{\varepsilon}_0]^T \quad [15.24]$$

and the corresponding vector of the nominal accelerations is obtained by derivation of [15.24]:

$$\ddot{q}_0 = \frac{d \dot{q}_0}{dt} \quad [15.25]$$

The main task of a robot controller is to ensure that the actual robot motion $q(t), t \in (0, T]$ converges to the nominal motion $q_0(t), t \in (0, T]$ as well as to enable satisfactory dynamic behavior of the robotic system even in the presence of system uncertainties and perturbations of different types.

15.5.2. Motion control

The control architecture of the 4WD robot rover (Figure 15.1) represents a modular hierarchy distributed structure (Figures 15.2 and 15.3). The proposed control architecture has two hierarchy levels – high level and low level [ROD 10b]. The high control level consists of a cognitive block (the fuzzy motion planning module, described in section 15.5.1) coupled with the complementary model-based dynamic controller. The controller takes into account dynamics of the entire system including robot rigid-body dynamics, tire nonlinear dynamics and dynamics of rover actuators described in equations [15.7–15.21]. The high control level block is charged with sensor-data acquisition and sensor-data fusion, signal processing, navigation and motion planning, control of robot dynamic behavior as well as distribution of control commands per particular robot wheels. A rover servo-control (low-level control) ensures appropriate power-driving according to the high-level control commands and driving conditions. The control system considered in

this study has a “distributed” character due to the reasons explained earlier. The low control variables represent corresponding tire torques and tire rpm (angular velocities) determined according to the actual tire–ground conditions as well as according to the referent robot motion. In such a way, the control system designed “takes care” about robot dynamics under real (variable) driving conditions.

The controller was designed under the following assumptions:

1) The model presented in this chapter by equations [15.1–15.21] describes the system’s physical behavior, with satisfactory accuracy being the main physical effects taken into account.

2) Parameters of the model can be determined in a precise way. The parameters are acquired from the CAD model or directly from the system by measurement (mass, moments of inertia, geometry parameters, etc.) or by estimation using corresponding sensorial information, a model of the system and a Kalman filter.

3) Corresponding tire–ground interaction parameters (slipping and rolling resistance coefficients) can be identified with satisfactory precision by using the tire nonlinear empirical model [15.14–15.17] and corresponding measurement data regarding the tire torques and a tire rpm.

Taking into account the previous consideration, the high-level control algorithm has to ensure accurate path tracking of the reference trajectory (obtained from the fuzzy motion planning module) and fine dynamic performances of the system. The control algorithm capable of achieving these requirements is derived in the following form using the robot model [15.8]:

$$\begin{aligned}
 T_x &= H^{(1)}(q) \cdot [\ddot{x}_0 - k_d^{(1)} (\dot{x} - \dot{x}_0) - k_p^{(1)} (x - x_0)] + h_{cgg}^{(1)}(q, \dot{q}) - F_w^{(1)}(q) \\
 T_y &= H^{(2)}(q) \cdot \ddot{y} + h_{cgg}^{(2)}(q, \dot{q}) - F_w^{(2)}(q) \\
 T_z &= H^{(3)}(q) \cdot \ddot{z} + h_{cgg}^{(3)}(q, \dot{q}) - F_w^{(3)}(q) \\
 T_e &= H^{(4)}(q) \cdot [\ddot{\epsilon}_0 - k_d^{(4)} (\dot{\epsilon} - \dot{\epsilon}_0) - k_p^{(4)} (\epsilon - \epsilon_0)] + h_{cgg}^{(4)}(q, \dot{q}) - F_w^{(4)}(q)
 \end{aligned} \tag{15.26}$$

where $k_p^{(i)}$ and $k_d^{(i)}$, $i = 1, \dots, 4$ are corresponding proportional and differential control gains in the particular directions (longitudinal and yaw) of motion. $H^{(i)}$, $h_{cgg}^{(i)}$ and $F_w^{(i)}$, $i = 1, \dots, 4$ are corresponding i th row, that is i th elements of H , h_{cgg} and F_w from model [15.8]. As has been explained in

section 15.3.1, the 4WD robot (Figure 15.1), as a non-holonomic system, is only able to perform a single displacement (forward or backward) and a single yaw rotation (turn). Thus, it can accomplish only two controlled DoF. Bearing this fact in mind, the robot (Figure 15.1) is controlled in an explicit way in the longitudinal (x) and yaw (ε) directions while in the other two coordinate directions (y and z) the rover is controlled implicitly [15.26]). The corresponding driving forces τ_x and torques τ_ε in the particular longitudinal and yaw directions are determined from:

$$\tau_x = T_x, \quad \tau_\varepsilon = T_\varepsilon \quad [15.27]$$

The driving forces τ_x and torques τ_ε , calculated in [15.27], are generated by the corresponding tire forces F_{xi} and F_{yi} , $i = 1, \dots, 4$ (Figure 15.5). In that sense, 4WD rover represents an “over controlled” system since there are four controlled tire forces F_{xi} , $i = 1, \dots, 4$ (lateral tire forces are not controlled in an explicit way) that should produce robotic motion in two controlled directions: longitudinal and yaw. The unknown tire forces F_{xi} can be determined from the system of equations [15.9] and [15.10] by imposing certain additional simplifications. Aiming at this, it is assumed that the right-side tires and the left-side tires (front and rear) rotate at the same speed (rpm). Through this assumption, the number of unknown variables F_{xi} , $i = 1, \dots, 4$ can be reduced twice. Assuming the previous assumption as well as the fact that the longitudinal forces F_{xi} , $i = 1, \dots, 4$ and the tire loads F_{zi} , $i = 1, \dots, 4$ correlate, the following auxiliary relations can be set as:

$$F_{x3} = \frac{F_{z1}}{F_{z3}} F_{x1}, \quad F_{x4} = \frac{F_{z2}}{F_{z4}} F_{x2} \quad [15.28]$$

The payload forces F_{zi} , $i = 1, \dots, 4$ are determined either directly by measurement at the robots tires (advance designs) or by calculation, taking into account a relative position of the robot body MC with respect to the geometry center C (Figure 15.4), and implementing a matrix of transformation regarding the case when the robot moves on the sloped surface. From relations [15.9] and [15.10], taking into account simplification [15.28] as the auxiliary relation, the system of two linear equations is derived. It enables determining the unknown tire forces F_{xi} , $i = 1, \dots, 4$. The mentioned system of linear equations has the form:

$$\begin{aligned}
\tau_x &= F_{x1} \left(1 + \frac{F_{z1}}{F_{z3}} \right) + F_{x2} \left(1 + \frac{F_{z2}}{F_{z4}} \right) \\
\tau_e &= F_{x1} \left(1 + \frac{F_{z1}}{F_{z3}} \right) \left(\frac{t_r}{2} + l_y \right) - F_{x2} \left(1 + \frac{F_{z2}}{F_{z4}} \right) \left(\frac{t_r}{2} - l_y \right) + A_e \quad [15.29] \\
A_e &= (F_{y1} + F_{y2}) \cdot (l_f - l_x) - (F_{y3} + F_{y4}) \cdot (l_r + l_x)
\end{aligned}$$

where A_e is a residual term that includes actual values of the lateral tire forces F_{yi} , $i=1, \dots, 4$. The searched variables (F_{xi} , $i=1, \dots, 4$) are used afterward for calculation of the particular tire speeds (rpm) ω_i , $i=1, \dots, 4$. For this purpose, the empirical tire model defined by relations [15.14–15.16] is utilized. Prior to determination of the tire rpm, the corresponding tire slip ratios s_i [15.3] and tire slip angles α_i [15.4] have to be identified that enable the final estimation of the controlled values of the tire speeds ω_i , $i=1, \dots, 4$.

Following the unique procedure of control variables calculation, tire torques τ_i (torques at the gear box shaft) are calculated by using [15.17]. Then, the reference values of the servo-controller are defined, using actuator model equations [15.18–15.21] and including the variables (tire torque τ_i and tire angular speed ω) calculated earlier. The procedure of control variables calculation results in controlled voltage of the DC motors at the end. Navigation, motion planning and control algorithms proposed in this study, considered in section 15.5, are verified by utilization of the appropriate software simulator as well as experimentally using the open-structure wheel-based robotic platform presented in Figure 15.1.

15.6. Simulation and experimental testing

The open-structure mobile robotic platform considered in this chapter is experimentally tested within the context of a networked system. Wireless robot-sensor networked (WRSN) systems refer to multiple robots operating together in coordination or cooperatively with sensors, embedded computers and human operators [MOR 08]. Cooperation entails more than one entity working toward a common goal while coordination implies a relationship between entities that ensures efficiency or harmony. Wireless communication between entities and advanced sensing are fundamental to both cooperation and coordination and hence the central role of the networked system. Networked robots allow multiple robots and auxiliary entities to perform

tasks that are well beyond the abilities of a single robot. Tasks such as searching or mapping, in principle, are performed faster with an increase in the number of robots. Perhaps the biggest advantage of using the network to connect robots is the ability to connect and harness physically removed assets. Mobile robots can react to information sensed by other mobile robots in the next room.

Bearing in mind the enormous importance of building and use of such WRSN systems, the necessities of accurate modeling, appropriate experimental testing and verification are unavoidable steps in design and development of such kinds of complex systems and their robot entities. One such system, developed in laboratory conditions [ROD 11], is used for experimental testing of navigation, motion planning and control algorithms considered in this chapter. A panoramic view to the experimental WRSN system developed is presented in Figure 15.8. Prior to the experimental validation of the algorithms designed and considered in this chapter, corresponding simulation experiments are needed in order to enable initial tuning of the system, testing of dynamic characteristics and analysis of system performances assuming the checking of the system robustness to different parametric and system uncertainties. In that sense, a model of an arbitrary, realistic multi-robot scenario, suitable for extensive simulation testing of different case studies regarding autonomous navigation, motion planning and control in the presence of obstacles (mobile and immobile), is used [ROD 10a]. The main screen presenting the graphical interface of modeling multi-robots scenario is shown in Figure 15.9.

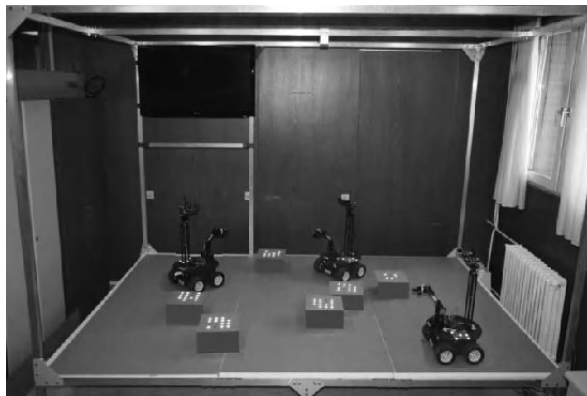


Figure 15.8. *Experimental test-bed station for research, development and benchmarking of motion planning and control algorithms designed for networked robotic systems*

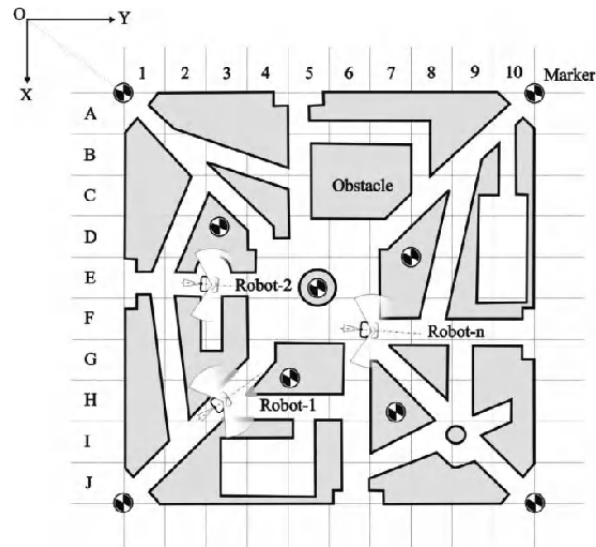


Figure 15.9. *Virtual WRSN – software simulator [ROD 10a], user-friendly graphical interface that enables imposing and simulation of multi-robot dynamic scenarios*

15.6.1. Case studies

In order to illustrate the proposed methodology of motion planning and control with a 4WD mobile robot (Figure 15.1), two typical case studies are analyzed: (1) motion of a single robot in the presence of stationary (immobile) obstacles, and (2) collision avoidance with mobile obstacles. Both scenarios are tested by using the simulator (Figure 15.9) as well as experimentally in a manner presented in Figure 15.8. In the considered cases, trajectory tracking accuracy as well as robot dynamic performances are evaluated.

The indoor mobile robot (Figure 15.1) is simulated. The parameters of the robot and tires were identified directly on the system by appropriate measurement or by estimation using the model defined in section 15.3. The parameters used in simulation are given in the Appendix. The robot is simulated moving in an unknown and unstructured environment (a labyrinth scenario, Figure 15.9) with uncertainty of tire–ground interaction (slippery surface). The imposed range of motion is assumed to be a square area of $15\text{ m} \times 15\text{ m}$ with the radial and transversal narrow corridors whose width is less than 1 m. The robot is required to move with a safe speed avoiding the stationary obstacles as shown in

Figure 15.9. The “Virtual WRSN”, a specialized Matlab®/Simulink toolbox, is used for simulation testing [ROD 10a].

The motion planning module (fuzzy block) and the corresponding dynamic controller are required to “find the shortest collision-free path” and to achieve desired motion in a way that ensures satisfactory dynamic behavior of the system. Characteristic simulation results, which contribute to the case-based analysis, are presented in Figures 15.10–15.13. The proposed algorithms of navigation and control ensure obstacle avoidance with a satisfactory precision even in an unknown and unstructured environment (Figure 15.10). The skid steering effects are shown in Figures 15.11 and 15.12. The longitudinal (traction/braking) tire forces are generated and changed independently on the robot wheels based on estimation of tire–ground adhesion parameters (using tire model and corresponding measurements of tire load). In such a way, lateral slipping and angular driftage of the robot are suspended (reduced to minimum) and this caused a smooth motion (no jerks). The smooth forward velocity profile and tire angular speeds (Figure 15.12) prove that the applied controller ensures reliable motion and fine dynamic behavior of the system. The control voltages of the DC motors are presented in Figure 15.13. The obtained voltages do not overcome the maximal range of 12 V. It proves that the control algorithm synthesized in this case fits well, both – the robot dynamics and also the tire–ground dynamics.

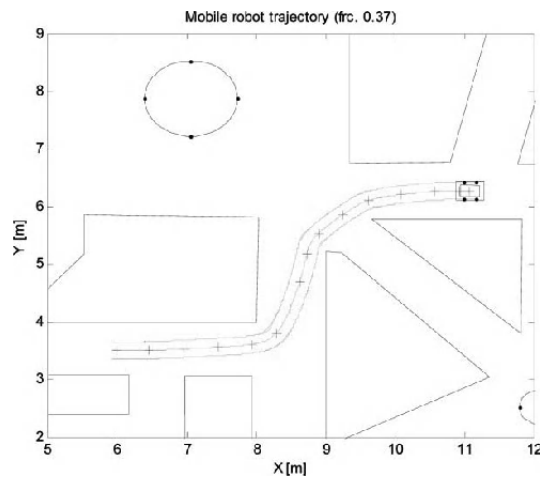


Figure 15.10. Example of the test scenario of obstacle avoidance in an unknown and unstructured environment using “Virtual WRSN” software simulator [ROD 10a]

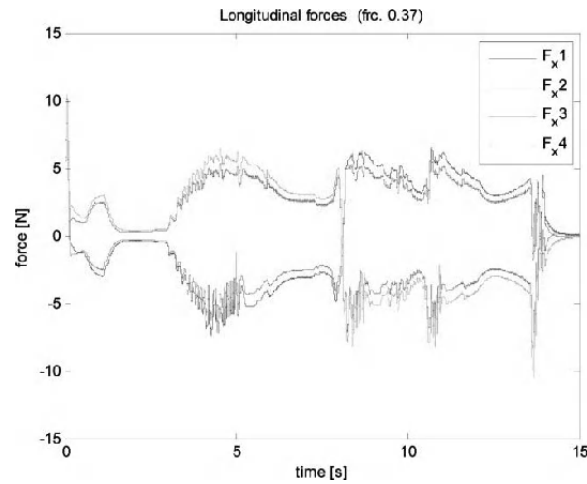


Figure 15.11. Longitudinal tire forces achieved during the avoiding scenario (Figure 15.10)

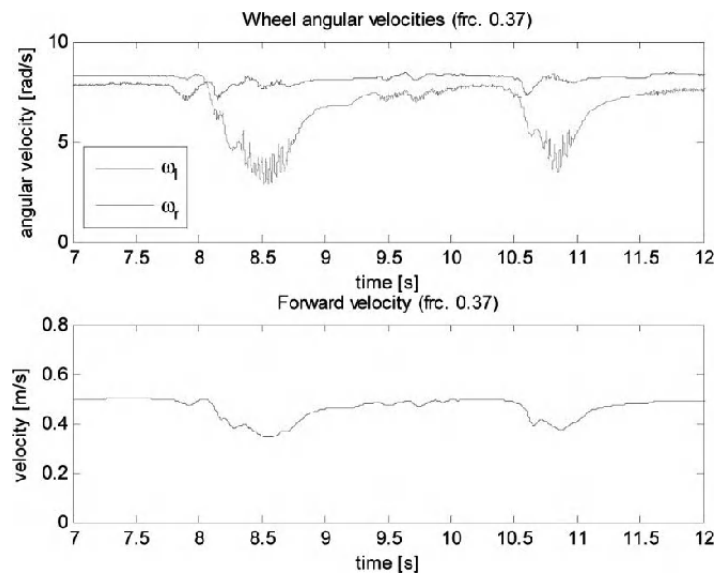


Figure 15.12. Actual forward speed and corresponding tire angular velocities (right and left side) achieved in the interval 7–12 s of a robot changing direction (“S” maneuver). For a color version of this figure please see www.iste.co.uk/mechatroneng.zip

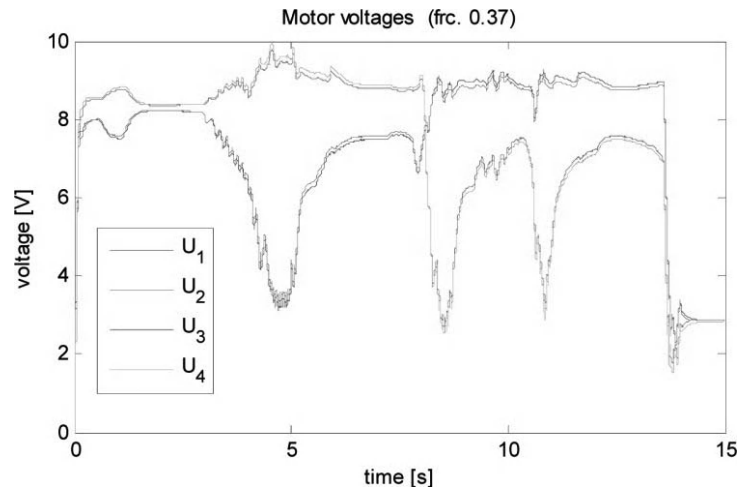


Figure 15.13. Control voltages of robot actuators (DC motors) generated to enable desired motion presented in Figure 15.10.

In the second case study, a dynamic multi-robot scenario of collision avoidance is simulated under the same conditions as in the previous case. Four robots with the same characteristics (kinematical and dynamic parameters) are used in the test. The robots are required to move independently in the small area intersecting their expected trajectories (Figure 15.14). The Coulomb friction coefficient of the ground surface is set to $\mu = 0.37$ as in the previous case study. The fuzzy motion planning modules of the particular robots have the tasks to identify the fastest collision-free trajectories and to enable reaching the target points with satisfactory precision. The concurrent robots are neither communicated between themselves nor in an indirect way by the basis station. The robots are restricted to use only their proximity and range sensors to detect mobile and immobile obstacles in their surroundings. The obtained simulation results are shown in Figures 15.14 and 15.15. The snapshots of the mobile robots in the particular characteristic sequences of motion are presented per second. From the plots shown in Figure 15.14, the way of obstacle avoidance among the moving robots can be observed by analyzing few characteristic time sequences. Particular robots, shown in the plots, stop and wait (Figure 15.15) for another robot(s) to pass away or they change the paths on-line to avoid collisions.

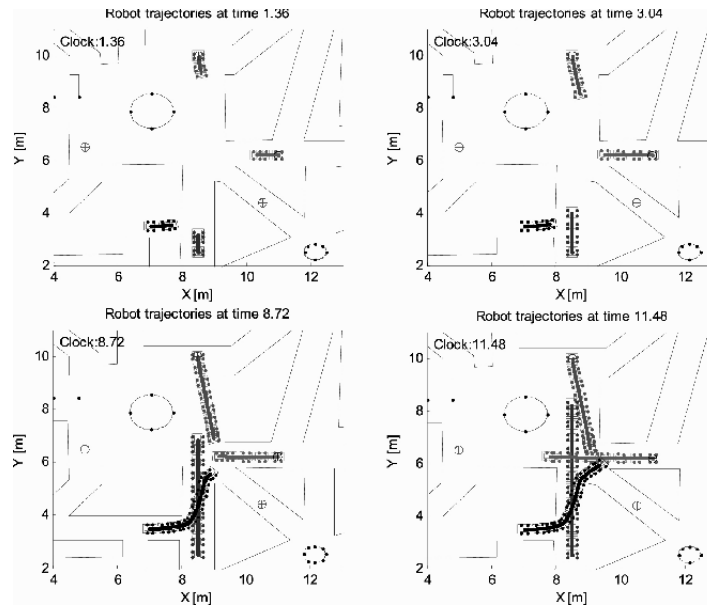


Figure 15.14. The snapshots of robots in a collision avoidance scenario captured at 1.38, 3.04, 8.72 and 11.48 s from the start of motion

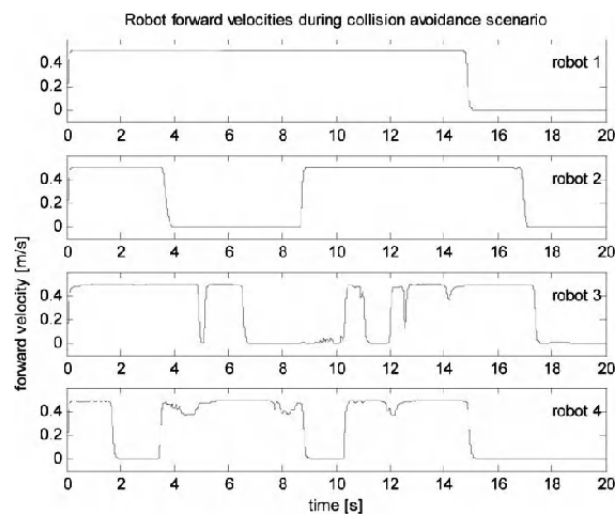


Figure 15.15. Time history plots of the forward speeds of the participating robots in the considered collision avoidance scenario presented in Figure 15.14.

15.7. Concluding remarks

The main contribution of this chapter is the development of an open-structure wheel-based mobile robotic platform and a complementary software simulator aimed at research, the development and education, as well as the objective benchmarking of this kind of advanced mechatronics system. The main objectives of the work are concerned with solving the problems of development of an autonomous navigation system, motion planning and control of mobile robots in unstructured environments in the presence of mobile and immobile obstacles and system uncertainties such as variation of tire-ground interaction performances. The described 4WD robotic system of an open structure as well as its extension to the wireless robot and sensor networked system (the unique test-bed station, Figure 15.8) offer great opportunities for research, analysis and objective benchmarking of such kinds robotic systems. Navigation, motion planning and control algorithms (including wireless communication) are of special interest for reliable and safe operation of the considered mechatronic system in the scope of an informatically structured environment. The complementary software simulator “Virtual WRSN” was also developed in order to support research and development and to speed up improvements of system performances. Both, the software simulator as well as the experimental test-bed station with the open-structure mobile robotic platform included represent useful research and engineering tools that can significantly simplify and speed up the designing process. Also, the system developed represents a scalable and a cost-effective mechatronic system whose architecture can be utilized for building advance industrial mobile wheel-based robotic platforms. Bearing in mind the aforementioned facts, advanced mechatronic systems of open structures, such as the system presented in this chapter, play a significant role in objective benchmarking and performance evaluation of various types of navigation and control algorithms designed for such robotic devices. Appropriate objective benchmarking and experimental system evaluation, under different unpredictable conditions of system operation, represent, today, the ultimate requirement for system performance optimization. This chapter represents a contribution in that sense.

The chosen case studies considered in this chapter have been used to illustrate the proposed methodology of motion planning and control of wheel-based robot dynamics in unknown environment with the disturbance of tire-ground parameters’ uncertainty type. The same robot (model and experimental system) was used for evaluation of different control algorithms aiming to achieve the best dynamic performances of the system in the

presence of mobile and immobile obstacles as well as in the case of the slippery surface. A corresponding performance analysis was done and the dynamic control (skid steering) is favored with respect to the kinematical control based on simplified differential steering approach. The drawbacks of the kinematical approach are the missing information about the robot dynamics and tire-ground interactions.

15.8. Acknowledgments

The study presented in this chapter represents results of the national research project “Ambientally intelligent service robots of anthropomorphic characteristics” TR-35003 supported by the Serbian Ministry of Education and Science and partially supported by the SNSF-SCOPES “CARE-robotics” project no. IZ74Z0_137361/1.

15.9. Bibliography

- [ADL 11] ADLS15PC – PC104-Plus Embedded single board computer, Advanced digital logic, 2011. Available at <http://www.adl-usa.com>.
- [AZA 94] AZARM K., SCHMIDT G., “Integrated mobile robot motion planning and execution in changing indoor environments”, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS'94*, Munich, Germany, pp. 298–305, 12–16 September 1994.
- [BOR 89] BORENSTEIN J., KOREN Y., “Real-time obstacle avoidance for fast mobile robots”, *IEEE Transaction on System Man and Cybernetics*, vol. 19, no. 5, pp. 1179–1187, 1989.
- [BOR 91] BORENSTEIN J., KOREN Y., “The vector field histogram – fast obstacle avoidance for mobile robots”, *IEEE Transaction on Robotics and Automation*, vol. 7, no. 3, pp. 278–288, 1991.
- [CAR 99] CARACCILO L., LUCA A.D., IANNITI S., “Trajectory tracking control of a four-wheel differentially driven mobile robot”, *Proceedings of the IEEE International Conference on Robotics and Automation ICRA'99*, Detroit, Michigan, pp. 2632–2638, 10–15 May 1999.
- [CHA 82] CHATILA R., “Path planning and environmental learning in a mobile robot system”, *Proceedings of the European Conference on Artificial Intelligence ECAI'82*, Orsay, France, pp. 112–118, 12–14 July 1982.
- [CHA 85] CHATTERGY R., “Some heuristics for the navigation of a robot”, *International Journal of Robotic Research*, vol. 4, no. 1, pp. 59–66, 1985.

- [DEV 11] DEVENTECH, ultrasonic range finder, 2011. Available at <http://www.robotshop.com>.
- [DFR 11] DFRobot, 4WD outdoor mobile platform, 2011. Available at <http://www.robotshop.com>.
- [FEI 94] FEITEN W., BAUER R., LAWITZKY G., "Robust obstacle avoidance in unknown and cramped environments", *Proceedings of the IEEE International Conference on Robotics and Automation ICRA'94*, San Diego, CA, pp. 2412–2417, 8–13 May 1994.
- [FIO 98] FIORINI P., SHILLER Z., "Motion planning in dynamic environments using velocity obstacles", *International Journal of Robotic Research*, vol. 17, no. 7, pp. 760–772, 1998.
- [FOX 97] FOX D., BURGARD W., THRUN S., "The dynamic window approach to collision avoidance", *IEEE Robotics and Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.
- [HAB 06] HABIB M.K., "Mechatronics engineering the evolution, the needs and the challenges", *Proceedings of the 32nd Annual Conference on IEEE Industrial Electronics Society IECON 2006*, Paris, France, pp. 4510–4515, 6–10 November 2006.
- [HAB 07] HABIB M.K., "Real time mapping and dynamic navigation for mobile robots", *International Journal of Advanced Robotic Systems*, vol. 4, no. 3, pp. 323–338, 2007.
- [KHA 86] KHATIB O., "Real-time obstacle avoidance for manipulators and mobile robots", *International Journal of Robotic Research*, vol. 5, no. 1, pp. 90–98, 1986.
- [KOZ 04] KOZŁOWSKI K., PAZDERSKI D., "Modeling and control of a 4-wheel skid-steering mobile robot", *International Journal of Applied Mathematics and Computer Science*, vol. 14, no. 4, pp. 477–496, 2004.
- [KRO 86] KROGH B.H., THORPE C.E., "Integrated path planning and dynamic steering control for autonomous mobile robots", *Proceedings of the IEEE International Conference on Robotics and Automation ICRA'86*, San Francisco, CA, pp. 1664–1669, 7–10 April 1986.
- [LUM 87] LUMELSKY V., STEPANOV A., "Path planning strategies for a point mobile automaton moving amidst unknown obstacles of arbitrary shape", *Algorithmica*, vol. 2, no. 4, pp. 403–430, 1987.
- [LYN 11] LYNXMOTION, AL5A 4 Degrees of freedom robotic arm Combo kit, 2011. Available at <http://www.lynxmotion.com>.
- [MAS 94] MASOUD A., MASOUD S., BAYOUMI M., "Robot navigation using a pressure generated mechanical stress field – the biharmonic potential approach", *Proceedings of the IEEE International Conference on Robotics and Automation ICRA'94*, San Diego, CA, pp. 124–129, 8–13 May 1994.

- [MAT 11] MATHWORKS, Matlab/Simulink, Fuzzy toolbox – user manual, 2011. Available at <http://www.mathworks.com/products/fuzzylogic>.
- [MIN 00] MINGUEZ J., MONTANO L., “Nearness diagram (ND) navigation: collision avoidance in troublesome scenarios”, *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 45–59, 2000.
- [MIN 04] MINGUEZ J., OSUNA J., MONTANO L., “A divide and conquer strategy to achieve reactive collision navigation systems”, *Proceedings of the IEEE International Conference on Robotics and Automation ICRA 2004*, New Orleans, LA, 26 pp. 412–418, April–May 2004.
- [MIN 05] MINGUEZ J., “The obstacle-restriction method for robot obstacle avoidance in difficult environments”, *Proceedings of the IEEE International Conference on Intelligent Robots and Systems IROS 2005*, Alberta, Canada, pp. 2284–2290, 2–6 August 2005.
- [MIN 08] MINGUEZ J., LAMIRAUX F., LAUMOND J.-P., “Motion planning and obstacle avoidance”, *Handbook of Robotics*, Springer, Berlin, pp. 826–850, 2008.
- [MIN 11] MINDSENSORS, infrared and light sensors, 2011. Available at <http://www.robotshop.com>.
- [MIT 10] MITROVIĆ S., ĐUROVIĆ Ž., “Fuzzy logic controller for bidirectional garaging of a differential drive mobile robot”, *Advanced Robotics*, vol. 24, nos. 8–9, pp. 1291–1311, 2010.
- [MOR 08] MORIN P., SAMSON C., “Motion control of wheeled mobile robots”, *Handbook of Robotics*, Springer, Berlin, pp. 729–825, 2008.
- [PAC 91] PACEJKA H.B., SHARP R.S., “Shear force development by pneumatic tires in steady state conditions: a review of modeling aspects”, *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, vol. 20, nos. 3–4, pp. 121–175, 1991.
- [PAC 93] PACEJKA H., BAKKER B., “The magic formula tire model”, *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, vol. 21, suppl. 001, pp. 1–18, 1993.
- [PAS 94] PASTERKAMP W.R., PACEJKA H.B., “On line estimation of tire characteristics for vehicle control”, *Proceedings of the International Symposium on Advanced Vehicle Control AVEC'94*, Tsukuba, Japan, pp. 521–526, 24–28 October 1994.
- [RAJ 78] RAJARAM S., MURUGESAN S., “A new method for speed measurement/control of DC motors”, *IEEE Transactions on Instrumentation and Measurement*, vol. 27, no. 1, pp. 99–102, 1978.
- [ROD 02] RODIĆ A., VUKOBRATOVIĆ M., *Dynamics, Integrated Control and Stability of Automated Road Vehicles*, Ibidem Verlag, Stuttgart, 2002.

- [ROD 10a] RODIĆ A., MESTER G., “Virtual WRSN – modeling and simulation of wireless robot-Sensor networked systems”, *Proceedings of the 8th International Symposium on Intelligent Systems and Informatics SISY 2010*, Subotica, Serbia, pp. 115–120, 10–11 September 2010.
- [ROD 10b] RODIĆ A., ADDI K., JEZDIMIROVIĆ M., “Sensor-based intelligent navigation and control of autonomous mobile robots in advanced terrain missions”, *Scientific Technical Review*, vol. 60, no. 2, pp. 7–15, 2010.
- [ROD 11] RODIĆ A., JOVANOVIĆ M., POPIĆ S., *et al.*, “Scalable experimental platform for research, development and testing of networked robotic systems in informationally structured environments”, *IEEE Symposium Series on Computational Intelligence SSCI 2011*, Paris, France, pp. 136–143, 11–15 April 2011.
- [RYU 11] RYU J.C., AGRAWAL S.K., “Differential flatness-based robust control of mobile robots in the presence of slip”, *The International Journal of Robotics Research*, vol. 30, no. 4, pp. 463–475, 2011.
- [SIM 96] SIMMONS R., “The curvature-velocity method for local obstacle avoidance”, *Proceedings of the IEEE International Conference on Robotics and Automation ICRA’96*, Minneapolis, MN, pp. 3375–3382, 22–28 April 1996.
- [SIN 96] SINGH L., STEPHANOU H., WEN J., “Real-time robot motion control with circulatory fields”, *Proceedings of the IEEE International Conference on Robotics and Automation ICRA’96*, Minneapolis, MN, pp. 2737–2742, 22–28 April 1996.
- [SME 99] SMEATON R.W., *Switchgear and Control Handbook*, 3rd ed., McGraw Hill, New York, 1999.
- [STA 11] HAGISONIC, StarGazer – infrared localization system, 2011. Available at <http://www.hagisonic.com/>.
- [STR 84] STRATEGIESL L., “Strategies for solving collision-free trajectories problems for mobile robots and manipulator robots”, *International Journal of Robotic Research*, vol. 3, no. 4, pp. 51–65, 1984.
- [SUR 11] Surveyor Stereo Vision System, Surveyor Corporation, 2011. Available at <http://www.surveyor.com>.
- [WAR 08] WARD C.C., IAGNEMMA K., “A dynamic-model-based wheel slip detector for mobile robots on outdoor terrain”, *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 821–831, 2008.
- [WIK] WIKIPEDIA, Ackermann steering geometry, http://en.wikipedia.org/wiki/Ackermann_steering_geometry
- [YI 07] YI J., SONG D., ZHANG J., *et al.*, “Adaptive trajectory tracking control of skid-steered mobile robots”, *Proceedings of the IEEE International Conference on Robotics and Automation ICRA 2007*, Rome, Italy, pp. 2605–2610, 10–14 April 2007.

15.10. Appendix

In addition to the study presented in the chapter, parameters used in simulation of case studies are presented in Table A1.1. Values of parameters are taken from the product sheet, or estimated by use of the model.

Parameter	Symbol	Value	Unit
Wheel radius	r_t	0.06	m
Front wheel distance	l_f	0.082	m
Vehicle track	t_r	0.29	m
Rear wheel distance	l_r	0.1	m
MC displacement in x	l_x	0.018	m
MC displacement in y	l_y	0	m
Mass	m	6.8	kg
Moment of inertia	I_z	0.0921	kgm ²
Gravity	g	9.81	m/s ²
Air resistance in x	K_x	1.36	Ns ² /m ²
Air resistance in y	K_y	1.50	Ns ² /m ²
Rolling friction	f_{ri}	0.02	—
Tire model coefficient	B	0.1884	—
Tire model coefficient	C	1.30	—
Tire model coefficient	D	38.84	N
Tire model coefficient	E	-2.8556×10^5	—
Rotor inductance	L_r	94×10^{-6}	H
Rotor resistance	R_r	2.35	Ω
Rotor inertia	J_r	50×10^{-9}	kgm ²
Viscous friction	B_c	1.1×10^{-6}	Ns/m
Reduction gear	N	67	—
Voltage constant	C_M	15×10^{-3}	Vs/rad
Torque constant	C_E	15×10^{-3}	Nm/A

Table A1.1. Parameters of 4WD rover, and their values used in simulation of case studies

Chapter 16

Design and Physical Implementation of Holonomous Mobile Robot – Holbos

16.1. Introduction

Development and miniaturization of electronic components as well as the development of new control algorithms made a great impact on robotics. Robots being built today are becoming even more robust and multifunctional, making them a partner for the future in a number of industries, in medicine, transport and in commercial branches of human activity [BUR 98, DWI 12, FLI 11, MCM 06]. In parallel with the improvement of technologies used in the production and the processes industry, as well as in transport and logistics, the automation and improvement of these processes became an imperative, aiming at larger production rates, but also at better quality products and the reduction in the number of spoiled products. The human error factor is the main reason for these problems occurring in industry today, so the use of robots and robotic systems seems to be an obvious solution. The “3D jobs” (dirty, dangerous and demeaning or dirty, dangerous and difficult) concept, which first emerged as a term that describes the three types of job unwanted by human workers, quickly became a guideline for application-based development of robots and robotic systems [AYR 81, WAL 08, ZAM 11].

Chapter written by Jasmin VELAGIC, Admir KAKNJO, Faruk DAUTOVIC, Muhidin HUJDUR and Nedim OSMIC.

The transport of heavy mechanical and machine constructions and materials in industry is one of the areas in which the application-based development of robots has been quite noticeable in last few decades [FAR 12, FLI 11]. By using pneumatic or hydraulic lifting systems installed on mobile robotic platforms (autonomous or remotely controlled), the lifting and transport tasks became easier and cheaper. A variety of solutions in this area that are already commercially available were thoroughly examined, which led to the main goal of this work – an implementation of a small-scale robotic platform for material or mechanical construction transport with the intention of integration of already existing solutions. Some new solutions will also be introduced [KUR 06].

Most of the existing solutions include remotely controlled robots that incorporate mecanum wheels allowing the omnidirectional movement. The use of these robots still includes the manual control and allows the robot to be used only on flat terrains. The robot being described in this chapter is addressing these issues while incorporating technical ideas and solutions already being used. The guidelines for the development of our robot are as follows:

- The mobile robot will use the omnidirectional movement technology based on mecanum wheels allowing the robot to be used on both flat and tilted terrains.
- The robot will have a retractable platform for transport with leveling option.
- The mobile robot will incorporate manual control based on ZigBee wireless technology, but will also allow the implementation of algorithms for local or centralized autonomous movement of a robot.
- The ultrasonic sensors, encoders and camera vision will provide the input data for algorithms for autonomous movement of a robot.

The locomotion, mechanical and electrical designs, and implementation of the proposed holonomous robot Holbos will be described in the following sections.

16.2. Locomotion of holonomous mobile robot

One of the main features of the holonomous robot is its capability to move in any direction, regardless of its orientation. In other words, the heading and the course of the robot do not need to achieve the same values.

The ability to start moving in any direction represents one of the advantages for this type of design, considering the fact that control for this type of system can be designed in a relatively simple manner [VEL 12]. There are two demands that are set to build the robot with the described capabilities, and both are related to the wheels. The locomotion system that would make this type of robot's movements possible needs to be realized using mechanically complex wheels. Independent speed and direction control for each wheel is another demand in the design of this type of robot. The wheels that are used on this robot are described below.

One type of the mechanically complex wheels that makes the robot holonomous is called mecanum wheel [BLA 90]. Each of these wheels has 12 rubber rollers that are mounted on the edge of the wheel and can rotate freely. The rubber rollers are mounted on the separate shafts that have an angle of 45° relative to the axis of wheel rotation. When the wheel rotates, the point of touch with the ground smoothly moves from one roller to another. Two existing types of these wheels, left-oriented and right-oriented, are always mounted in pair on one shaft of the robot. The way in which the wheels are mounted on a robot is shown in Figure 16.1.

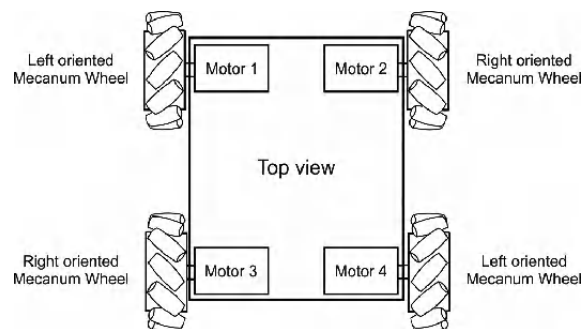


Figure 16.1. *Mecanum wheels and motors on robot*

When all wheels rotate in one direction, clockwise or counterclockwise, the robot is moving forward or backward, respectively. If the wheels on the left side of the robot (according to Figure 16.1) rotate in one direction and the wheels on the right side of the robot rotate in another direction then the robot will rotate in clockwise or counterclockwise direction, depending on the direction of rotation of the wheels. If the wheels in the opposite corners of the robot have the same direction of rotation (wheels one and four rotate in one direction, while wheels two and three rotate in a direction which is opposite

to the direction of rotation of wheels one and four), the robot will move to the left or right side, depending on the direction of rotation of wheel pairs. If the wheels in opposite corners of the robot rotate in one direction, while the other two wheels do not rotate, the robot will move diagonally in the direction that is dependant on the direction of rotation of the two wheels. The principle described above is shown in Figure 16.2.

After the dependency of the direction of the robot movement on the mecanum wheel direction of rotation is explained, it is possible to derive a mathematical model of the robot.

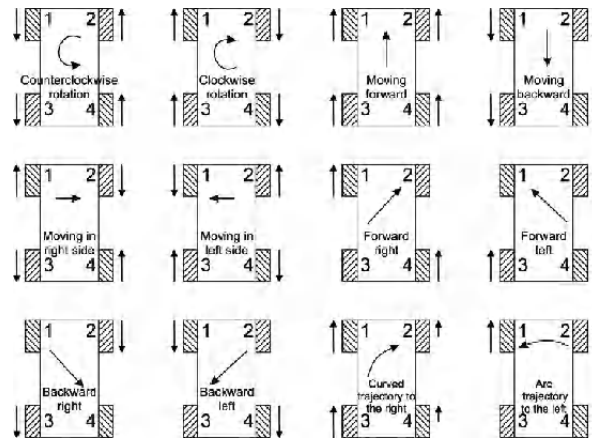


Figure 16.2. Examples of directions of the movement of the robot

16.2.1. Kinematic model of robot

A kinematic model of holonomous mobile robot in coordinate system xOy (Figure 16.3) can be expressed using the following equation [VEL 12]:

$$x(k+1) = x(k) + v_x \Delta t \quad [16.1]$$

$$y(k+1) = y(k) + v_y \Delta t \quad [16.2]$$

$$\varphi(k+1) = \varphi(k) + \omega \Delta t \quad [16.3]$$

Now, the x and y components of the linear velocity of the robot as well as the angular velocity of the robot need to be determined in every time

instant k . The key point in determining those values is the equation for linear velocity of every wheel in the coordinate system that is attached to the robot.

The linear velocity of each wheel in a coordinate system attached to the robot can be expressed as [TLA 08]:

$$v_i' = \text{sgn}(\dot{\theta}_i) K_i r \dot{\theta}_i \quad [16.4]$$

where K_i is the wheel coefficient that depends on the number of the rollers on the wheel and the distance between two neighbor rollers, r is the diameter of the wheel and $\dot{\theta}_i$ is the angular velocity of i th wheel.

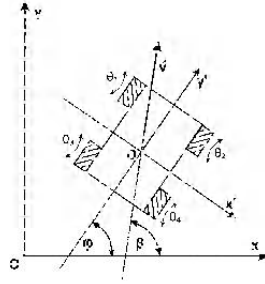


Figure 16.3. Global and local coordinate system of the robot [TLA 08]

It is now possible to decompose the linear velocity of the wheels (equation [16.4]) along x' and y' axis of the $x'Oy'$ coordinate system, which yields:

$$v_{ix}' = \text{sgn}(\dot{\theta}_i) K_i r \dot{\theta}_i \cos(\alpha) \quad [16.5]$$

$$v_{iy}' = \text{sgn}(\dot{\theta}_i) K_i r \dot{\theta}_i \sin(\alpha) \quad [16.6]$$

where α represents the angle of the rollers relative to the axis of rotation of the wheel.

The total linear velocity components of a mobile robot along x' and y' axis of the $x'Oy'$ coordinate system are sums of velocity components for all wheels. The total linear velocities along x' and y' axis can be expressed as:

$$v'_x = \sum_{i=1}^4 v'_{ix} = \sum_{i=1}^4 \operatorname{sgn}(\dot{\theta}_i) K_i r \dot{\theta}_i \cos(\alpha) \quad [16.7]$$

$$v'_y = \sum_{i=1}^4 v'_{iy} = \sum_{i=1}^4 \operatorname{sgn}(\dot{\theta}_i) K_i r \dot{\theta}_i \sin(\alpha) \quad [16.8]$$

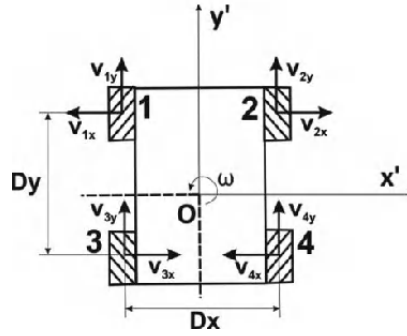


Figure 16.4. *Velocities of the wheels*

It is necessary to transform equations written in the coordinate system $x'O'y'$ into equations in global coordinate system xOy . In this case, the linear velocity along the x axis in the global coordinate system xOy is described by:

$$\begin{aligned} v_x &= \sin(\varphi) \sum_{i=1}^4 v'_{ix} + \cos(\varphi) \sum_{i=1}^4 v'_{iy} = \\ &= \sin(\varphi) \sum_{i=1}^4 \operatorname{sgn}(\dot{\theta}_i) K_i r \dot{\theta}_i \cos(\alpha) + \cos(\varphi) \sum_{i=1}^4 \operatorname{sgn}(\dot{\theta}_i) K_i r \dot{\theta}_i \sin(\alpha) = \\ &= \sin(\varphi) K r \cos(\alpha) [\dot{\theta}_1 - \dot{\theta}_2 - \dot{\theta}_3 + \dot{\theta}_4] + \cos(\varphi) K r \sin(\alpha) [\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3 + \dot{\theta}_4] \end{aligned} \quad [16.9]$$

The equation for the linear velocity along axis y in the global coordinate system xOy can be expressed as:

$$\begin{aligned} v_y &= \cos(\varphi) \sum_{i=1}^4 v'_{ix} + \sin(\varphi) \sum_{i=1}^4 v'_{iy} = \\ &= \cos(\varphi) \sum_{i=1}^4 \operatorname{sgn}(\dot{\theta}_i) K_i r \dot{\theta}_i \cos(\alpha) + \sin(\varphi) \sum_{i=1}^4 \operatorname{sgn}(\dot{\theta}_i) K_i r \dot{\theta}_i \sin(\alpha) = \\ &= \cos(\varphi) K r \cos(\alpha) [\dot{\theta}_1 - \dot{\theta}_2 - \dot{\theta}_3 + \dot{\theta}_4] + \sin(\varphi) K r \sin(\alpha) [\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3 + \dot{\theta}_4] \end{aligned} \quad [16.10]$$

The total linear velocity v of the robot can be expressed as a sum of the velocities v_x and v_y from relations [16.9] and [16.10]:

$$\vec{v} = \vec{v}_x + \vec{v}_y \quad [16.11]$$

The direction of the robot can be written as:

$$\beta = \arctg\left(\frac{v_x}{v_y}\right) \quad [16.12]$$

To determine an angular velocity of a mobile robot, all velocity components that contribute to the angular velocity have to be taken into consideration. For that reason, it is convenient to split all the wheels of the robot into four groups, as shown in Figure 16.4. The robot has non-zero angular velocity if there is any difference in velocity between the wheels on left and on the right side (if the robot is split along the y axis), or if there is any difference in velocity between the wheels on the top and on the bottom side (if the robot is split along the x axis). Consequently, the contribution to angular velocity from every group of wheels is described as follows:

$$\begin{aligned} v_{yL} &= v_{1y} + v_{3y} & v_{xD} &= v_{3x} - v_{4x} \\ v_{yR} &= v_{2y} + v_{4y} & v_{xD} &= -v_{1x} + v_{2x} \\ \Delta\theta_y &= \frac{v_{yL} - v_{yR}}{Dx} & \Delta\theta_x &= \frac{v_{yD} - v_{yU}}{Dy} \\ \omega = \Delta\theta &= \Delta\theta_x + \Delta\theta_y = \frac{v_{yL} - v_{yR}}{Dx} + \frac{v_{yD} - v_{yU}}{Dy} \end{aligned} \quad [16.13]$$

The kinematic model of a holonomous mobile robot can now be expressed as:

$$\begin{vmatrix} x(k+1) \\ y(k+1) \\ \varphi(k+1) \end{vmatrix} = \begin{vmatrix} x(k) + v \cos(\beta) \Delta t \\ y(k) + v \sin(\beta) \Delta t \\ \varphi(k) + \omega \Delta t \end{vmatrix} \quad [16.14]$$

Equation [16.14] can be rewritten in this form:

$$\begin{aligned} x(k+1) &= x(k) + \\ &\left\{ s \sin(\varphi) K r \cos(\alpha) [\dot{\theta}_1 - \dot{\theta}_2 - \dot{\theta}_3 + \dot{\theta}_4] + \cos(\varphi) K r \sin(\alpha) [\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3 + \dot{\theta}_4] \right\} \Delta t \end{aligned} \quad [16.15]$$

$$y(k+1) = y(k) + \left\{ \cos(\varphi)Kr \cos(\alpha) \left[\dot{\theta}_1 - \dot{\theta}_2 - \dot{\theta}_3 + \dot{\theta}_4 \right] + \sin(\varphi)Kr \sin(\alpha) \left[\dot{\theta}_1 + \dot{\theta}_2 + \dot{\theta}_3 + \dot{\theta}_4 \right] \right\} \Delta t \quad [16.16]$$

$$\varphi(k+1) = \varphi(k) + \left(\frac{\text{sgn}(\dot{\theta}_1)\dot{\theta}_1 + \text{sgn}(\dot{\theta}_3)\dot{\theta}_3 - \text{sgn}(\dot{\theta}_2)\dot{\theta}_2 - \text{sgn}(\dot{\theta}_4)\dot{\theta}_4}{Dx} + \frac{\text{sgn}(\dot{\theta}_1)\dot{\theta}_1 + \text{sgn}(\dot{\theta}_3)\dot{\theta}_3 - \text{sgn}(\dot{\theta}_2)\dot{\theta}_2 - \text{sgn}(\dot{\theta}_4)\dot{\theta}_4}{Dy} \right) \Delta t \quad [16.17]$$

Equations [16.15]–[16.17] represent the kinematic model of an holonomous mobile robot. In the following section, a mechanical design of the proposed holonomous mobile robot will be described.

16.3. Mechanical design

Since the robot building is done on a small scale and the accent was not on the mechanical design, the decision was made to use a commercially available plastic box with a reinforced bottom. This made the robot durable, but at the same time lighter. It also allowed easier mounting of other mechanical and electronic parts. In the four corners of the lower part of the mobile construction four motors were mounted. The mecanum wheels were directly mounted onto the motors, together with encoder sensors allowing the measurement of the speed of each of the wheels. Inside the mobile robot, a 12 V battery together with the printed circuit boards was mounted. Four linear actuators were also mounted in the four corners of the lower part of robot. The upper part of the robot, which can be easily disconnected in order to access the interior of the robot, is used as a platform. Although the main purpose of this platform was to implement and test the platform control algorithms and not to lift heavy objects, it turned out that it allowed a significant load to be lifted and carried (around 5 kg). On the bottom side of the platform, at its center, an inertial measurement unit (IMU) was mounted, allowing the sensing of the tilt angle of the platform. Four light-emitting diodes were also added at corners of the platform. These light-emitting diodes blink when the platform is being manually controlled. The ultrasonic sensors were mounted on a steel support mounted on the robot. These sensors were installed in 4-2-2-4 formation (four on the front and back and two on the left and right sides), sensing the robots distance from the obstacle. The steel support for the kinect camera was also mounted on the back side. This makes the development, testing and use of complex, vision-based, robot

control algorithms possible on this robot. The three-dimensional model of the robot and its picture are shown in Figure 16.5.

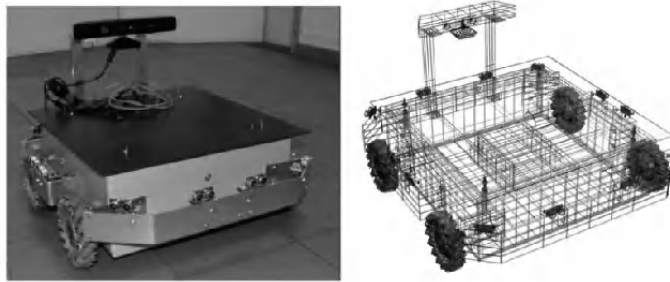


Figure 16.5. *Picture and 3D model of Holbos robot*

For the wireless manual control of the robot, the Sony PlayStation joystick was used. Together with the ZigBee wireless communication module and a display, it was connected with the Arduino Uno development board. The controller is powered by 9 V battery and is shown in Figure 16.6.



Figure 16.6. *Manual control module*

16.4. Electrical design

16.4.1. *Hardware components and software implementation*

The hardware structure of the robot is designed as a distributed control system. There are three separate modules that are responsible for controlling

different parts of the robot (different subsystems). These modules are interconnected using a RS-232 serial or the ZigBee wireless communication, depending on the chosen control module type. The block structure of the hardware system of the robot is shown in Figure 16.7. This section, which deals with the description of the modules, is divided into four parts as follows:

- main module (sensors and communication);
- manual control module (joystick);
- platform control module;
- motor control module.

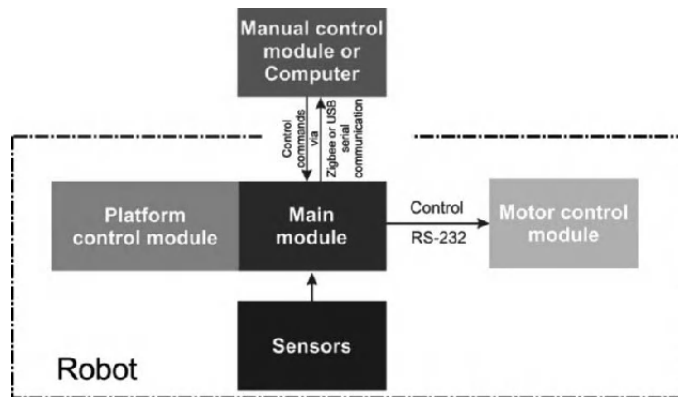


Figure 16.7. Block structure of realized system

This type of design allows for flexibility in terms of controlling the robot. There are two possible modes of robot operation: manual remote control and automatic control. To control the robot manually, a special joystick with ZigBee wireless communication was made. ZigBee wireless communication makes remote controlling of the robot possible. By placing the laptop computer, or some other piece of hardware with some control algorithms, on the robot platform, and connecting it to the main module, the robot can become autonomous.

16.4.1.1. Main module

The main module got its name because it represents the hardware device that interconnects all other modules on the robot. All sensors that are

mounted on the robot are also connected to this board. During the planning stage of the project, it was found that this device has to have at least four hardware modules for serial communication and a large number of digital and analog inputs. After reviewing several possible options, the decision was made to use an Arduino Mega board for this purpose. Not only does the board fulfill all the demands, but it also represents a low-cost solution for this type of use. With four serial communication modules and a total of 60 analog and digital inputs, the Arduino Mega board is an ideal solution for the main module of the robot.

The main module is responsible for several important tasks. The first task is to receive commands from the joystick or computer and to send them to the motor control module or to use them as control commands for the platform control module. These commands can be generated by a human operator using the joystick, or by a computer, and they make the manual control of the robot and platform movements possible. In the case of the computer control of the robot, the main module has to collect data from various sensors and send them back to the computer. On the basis of these data, the computer can run different algorithms and, based on that, different movement commands can be issued and sent to the main module. There are four types of sensors that are directly connected to main module:

- an ultrasonic range finder;
- an incremental encoder;
- a 2-axis gyroscope;
- a 3-axis accelerometer.

16.4.1.1.1. Ultrasonic range finder

There are 12 ultrasonic range finder sensors mounted on the robot body. Two sensors are mounted on each side, while four sensors are placed in the corners of the robot at an angle of 45° relative to the line perpendicular to any side of robot. The way that sensors are mounted on the edge of the robot is shown in Figure 16.8. Sensors are labeled with numbers – from one to 12. The numbers show the order in which the data from the sensors are collected (sensor number one is checked first, then sensor number two, etc.). Sensors can measure the distance from the objects in its environment in the range of 40–400 cm from the robot. The main module collects the distance data from every sensor and, if needed, sends them to the computer.

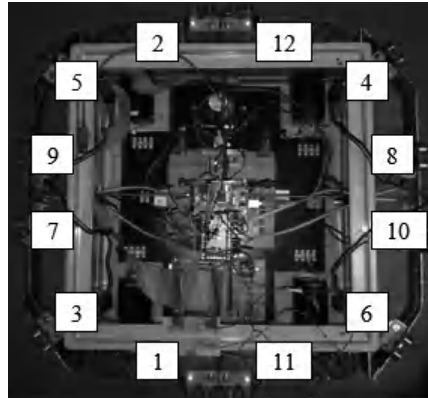


Figure 16.8. *Ultrasonic range finders on robot*

16.4.1.1.2. Incremental encoders

Each wheel on the robot is equipped with the incremental encoder. Signals from encoders are fed directly to digital inputs of the main module, where the frequency of every signal is calculated and, if needed, sent to the computer. The frequency of the encoder signal is directly proportional to the speed of the rotation of wheels. The data from encoders is very valuable in most applications where localization of the robot is needed.

16.4.1.1.3. Kinect camera

Kinect is a device that was developed by Microsoft for use in their game console Xbox 360. It has a Charge-Coupled Device (CCD) Red Green Blue (RGB) camera, 3D depth sensor and a microphone array. The possibility of acquiring both 2D and 3D images of the environment in the same time made this camera very convenient for use in robotic applications. The camera connects to a computer using a USB serial port, and if appropriate drivers are used, it is possible to get raw data from the camera.

16.4.1.1.4. Communication

Every message, that needs to be sent from one module to another, first has to be sent to the main module of the robot, which then forwards the message to its destination. Some hardware components that are used for communication were already mentioned in the previous sections. In this section, a detailed description of the messages format and the communication protocol will be provided.

The establishment of communication between the main module and the joystick or computer is the first task to be performed after turning on the robot. When the robot is switched on, the main module expects to receive an initialization message from the joystick. The initialization message consists of six bytes, of which the first byte is a hash sign “#”. After this sign, there are four zeros and letter “E”. When the message with this content is received by the main module, it replies with the same format message. These two messages differ only by the last character. The main module replies with the message that ends with character “O”, which means that communication is established or that communication is “OK”. When this message is received by the joystick, notification about successfully established communication appears on the LCD display, so the operator can see the result of the initialization stage. It is important to point out that the format of the message that the joystick sends to the main module is always same and has the following form: “#n1n2n3n4C”. Every message consists of the hash sign, four numbers (n1, n2, n3 and n4) and a character. In this section, the meaning of the various characters and messages that are sent will be explained.

Before the communication is further explained, it is important to say few things about the modes of operation of the robot. In the previous section, it was stated that the robot has two modes of operation, dependent on the device that controls the robot (joystick or computer). These modes of operation were called manual and automatic and they refer to the presence of the human operator. These two modes should not be mistaken with the modes of operation of the platform, which are also called automatic and manual. In the automatic mode of operation, the platform is controlled by the platform control module, whereas in the manual mode the platform is controlled by the human operator or computer. A detailed description of the way that the platform works will be provided in one of the following sections.

After initialization, the operator can choose between two modes of operation of the robot – control of the robot movements or platform control. The selection between these two modes of operation can be done by pressing the buttons “Start” or “Select” on the joystick. If the operator presses the “Start” button, the message that ends with the character “S” is sent to the main module, which means that the robot movement control mode is chosen. If the operator presses the “Select” button, the message that ends with the character “s” is sent to the main module, which means that the platform control mode is chosen. The mode selection message can be sent out at any time whilst operating the robot, when a change in the mode of operation is needed. The described algorithm can be written as pseudocode as shown in Table 16.2.

In the robot movement control mode, the message from the joystick carries four set-point values for the speeds of every wheel. The message ends with the character “A”. The speed set-points can be in a range [0, 255], where the range [0, 127] represents a positive direction and the range [128, 255] represents a negative direction of rotation of every wheel. This message can be graphically represented as shown in Figure 16.9. Four numbers that are sent in every message are ignored in every other mode of the robot operation, and the message is decoded using the last character. Different characters mean a different action.

Byte	1	2	3	4	5	6
Character	#	Speed1	Speed2	Speed3	Speed4	A

Figure 16.9. *Movement control mode message*

In the platform control mode of operation, there are four different messages that can be sent to the robot. The messages differ only by the last characters. Four numbers in every message are equal to zero. The messages for manual platform control are shown in Table 16.1, together with the explanation of every message.

Message	Meaning
#0000G	Raise the platform
#0000P	Lower the platform
#0000R	Raise one side of the platform
#0000B	Raise other side of the platform

Table 16.1. *Messages for manual platform control*

In the case of computer control of the robot, a communication protocol can be easily expanded in the way that the main module sends the sensor data to the computer. When the computer is placed on the platform, it connects to the main module using a USB cable, which allows full duplex communication between computer and robot. With full access to sensor data, various advanced control algorithms can be realized and the robot can be turned into fully autonomous vehicle.

Startup – program	
1	while “Initialization” message is not received
2	Determine if message is “Initialization” message
3	end while
4	Send message: “Communication established”
5	while “Mode selection” message is not received
6	Read message
7	Determine if message is “Mode selection” message
8	end while
9	Determine the mode of operation
10	if mode of operation is “Control of robot movements”
11	Control_of_robot_movements()
12	end if
13	if mode of operation is “Platform control”
14	Platform_control()
15	end if
16	End

Table 16.2. Algorithm that runs when the system is powered

16.4.1.2. Manual control module (joystick)

As stated in the previous section, a robot has two modes of operation, manual and automatic. In automatic mode, the robot is in control of a computer and can perform tasks autonomously. However, automatic operation is an extension to the original design of the robot, which is primarily designed to be operated by human operator. Thus, the robot is equipped with a device that makes remote operation by a human possible.

Some of the goals that were set for the joystick design, in addition to the possibility of remote operation, were the intuitive control and ergonomic design of the joystick. For these reasons, it was decided to use the type of joystick that is typical for game consoles, such as PlayStation. The joystick is equipped with the Arduino Uno board and ZigBee module for wireless communication. The Arduino board receives messages from the joystick and transforms them into instructions that are sent to the main module of the robot. The modes of operation of the robot and some other important messages are shown on an LCD display that was also mounted on the

joystick and connected to the Arduino Uno board. The block structure of the manual control module is shown in Figure 16.10. The algorithm, described in words, is shown as pseudocode in Table 16.3.

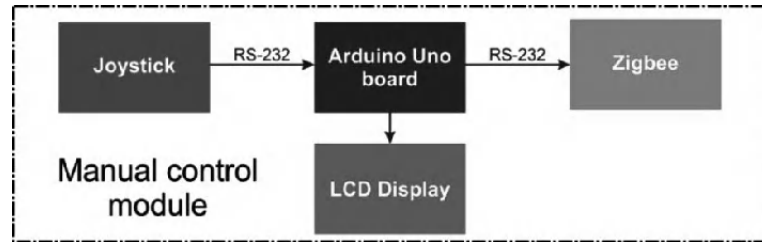


Figure 16.10. Structure of manual control module

Joystick_Algorithm	
1	while “Acknowledge message” is not received
2	Send the “Initialization message”
3	end while
4	Display “Communication established” text on LCD display
5	while 1
6	if operator pressed the button
7	if “Change the mode of operation” message sent
8	Change text on display
9	end if
10	Send the message to main module
11	end if
12	end while
13	End

Table 16.3. Algorithm that runs in manual control module

16.4.1.3. Motor control module

Mechanical realization of the locomotion system was explained in detail in previous two chapters. This section deals with the hardware realization of a control module for motors. As stated in previous chapters, the robot has four wheels and four motors in total, with every wheel mounted directly on the shaft of the motor. This approach provides the ability to control every motor

separately and thus gives more flexibility when defining the robot's trajectory. The block structure of the module that is responsible for motor control is shown in Figure 16.11.

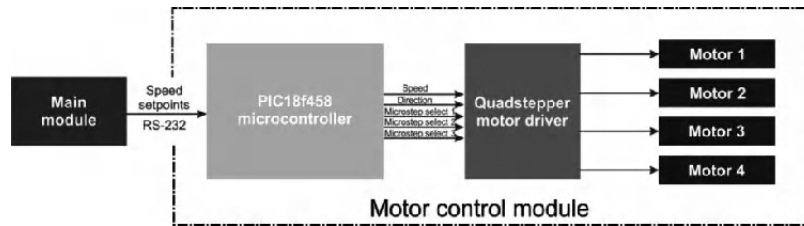


Figure 16.11. Motor control module

The module consists of two main components, the microcontroller-based board and the motor driver.

16.4.1.3.1. Microcontroller-based board

The main task of the microcontroller is to receive speed set-points for every motor, and to generate control signals for the driver board. Serial RS232 communication is used to transmit speed set-point values from the main module to a PIC18f458 microcontroller. Five signals need to be generated for each motor, a total of 20 signals for four motors, in order for the motor driver to work properly. Two of the five signals are speed and direction signals for each motor. The speed signal is in the form of squares with an amplitude of 5 V. The frequency of the squares is directly related to the speed of the motor. Every rising edge of the signal causes the motor to rotate by one step (1.8°) in a clockwise or an counterclockwise direction. The speed of each motor is directly proportional to the frequency of the control signal; the higher the frequency of the control signal, the higher is the speed of rotation of the motor. The direction signal dictates the direction of rotation. If the signal is in a low state, the direction is clockwise, and vice versa – if the signal is in a high state, the direction is counterclockwise.

16.4.1.3.2. Motor driver

For more simple control of the motors, a Quadstepper motor driver, and a single board with four motor drivers was used. As stated above, every driver requires five signals for every motor. The purpose of speed and direction signals is explained in the previous section. The other three signals (microstep select) are used to determine the mode of operation for the driver.

There are five different modes of driver operation (full step, half step, quarter step, eighth step and sixteenth step). Quarter, half and full step microstepping modes are used. When starting the rotation of a stepper motor, it is convenient to start with lower speeds. Quarter and half microstepping modes are used in this respect. The speed of the stepper motor can be lowered by choosing one of the microstepping modes of operation, while the frequency of the speed signal can remain constant and equal to the set-point value of the speed. When the stepper motor is starting, first the quarter step mode is on. After a quarter of a second, the mode changes to half step and after another quarter of a second the mode of operation becomes the full step mode. The change of the modes of operations ensures the rotation of the motor, even when the robot has to carry an increased amount of load.

The way in which the main module sends the set-points to the motor control module can be represented as shown in Table 16.4.

Control_of_robot_movements	
1	while “Change the mode of operation” message is not received
2	if message is received
3	Read the message
4	if message contains commands for motors
5	Send control message to Motor control module
6	end if
7	end if
8	end while
9	Control_of_robot_platform()
10	End

Table 16.4. Algorithm for controlling robot’s movements

16.4.1.4. Platform control module

The position of linear actuators and the IMU unit is described in section 16.3. The Firgelli L12-50-100-12-I multipurpose linear actuators were used. These actuators include the on-board controller that allows servo (0–5 V), pulse-width modulation (PWM) (Transistor-transistor logic (TTL) level signal) or current position (4–20 mA) control and have a stroke of 0–5 cm. The actuators output voltage signal (0–3.3 V) can be used to implement a closed-loop position control. The on-board controller is capable of discovering the used input control signal without any manual adjustments.

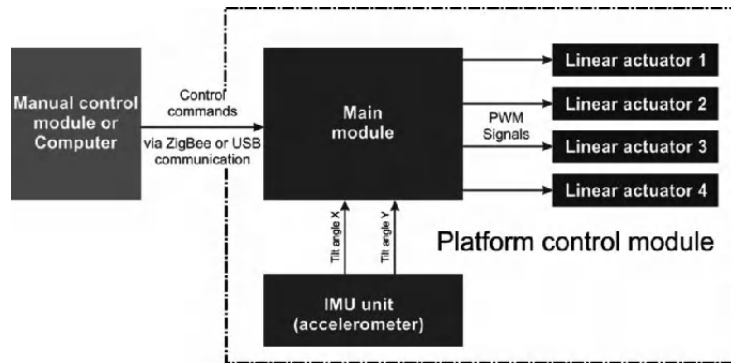


Figure 16.12. *Platform control module*

The IMU unit used for platform leveling is a five degree of freedom (DoF) unit that incorporates the IDG500 gyroscope and the ADXL335 accelerometer. This IMU unit, besides its intended use, can be used for localization algorithms when the platform is stationary by measuring roll, pitch and the tilt angle, as well as the X, Y, Z coordinates of the robot.

The platform control algorithm generates the PWM signals for each of the four linear actuators. These signals are TTL level square signals and have the frequency of 500 Hz. The stroke of 0–5 cm is corresponding to the 0–100% of PWMs duty cycle. Since the task was to keep the platform level, regardless of the tilt angle of the terrain the robot was moving on, the IMU unit was used to measure the tilt angle of the platform. Because of the nature and logic of this task, the exact data on tilt angle were not needed and since the linear accelerations of the robot in practice were negligible, the data given by the accelerometer were sufficient for the control implementation.

The block scheme of the platform control module is given in Figure 16.12. As can be seen in the figure, the control of platform movement is received via ZigBee wireless communication or the USB port of Arduino Mega unit. The command can be received from a remote controller or from any other unit with an on-board control algorithm that is connected to the main module. Based on commands received by the main module, which were already explained in the previous section, there are two modes of operation – robot movement control mode (selected by pressing the “start” button on the joystick) and the platform control algorithm mode (selected by pressing the “select” button on the joystick). In platform control algorithm mode, the movement of platform depends on the message that is sent from the user

interface (joystick or computer) to the platform. This mode is used only for manual control of the platform and can be written in pseudocode as shown in Table 16.5. When the robot is in robot movement mode, the platform leveling is autoregulated based on the readings from the accelerometer. The implemented algorithm is described by pseudocode as shown in Table 16.6.

Platform_Manual_Control	
1	while “Change the mode of operation” message is not received
2	if message is received
3	Read the message
4	if message contains commands for actuators
5	Send control message to Platform control module
6	end if
7	end if
8	end while
9	Control_of_robot_movements()
10	End

Table 16.5. Algorithm that runs when platform control module is in manual mode

In Table 16.6, the following abbreviations are used:

- DIV_X is digital value of the x component of acceleration.
- DIV_X0 is digital value of the x component of acceleration at 0° angle.
- DIV_Y is digital value of the y component of acceleration.
- DIV_Y0 is digital value of the y component of acceleration at 0° angle.
- SOA is sensitivity of accelerometers.

The power supply voltage of the accelerometer used is 5 V and its sensitivity is 0.33 V/g, where V is voltage and g is the relative gravitational acceleration. Constant 1,023 in denominators is the number of quantized levels of a 10 bit analog-digital converter. The equations for angle calculation are derived from a trigonometric description of the gravitational acceleration vector and its components, as can be seen in Figure 16.13.

Platform_Automatic_Control	
1	Get the digital value of the analog signals of current X and Y components of gravitational acceleration
2	Calculate the value of X and Y components of gravitational acceleration by using equations: $g_x = \frac{DIV\_X - DIV\_X0}{1023 \times SOA}$ $g_y = \frac{DIV\_Y - DIV\_Y0}{1023 \times SOA}$
3	Calculate the value of tilt angles (in degrees) by using equation: $angle\_X = \arcsin(g_x) \frac{180}{\pi}$ $angle\_Y = \arcsin(g_y) \frac{180}{\pi}$
4	if any of the angle values is greater than $\pm 1^\circ$
5	Choose which linear actuators are to be extracted based on angle values and signs
6	end if
7	Extract the chosen linear actuator by 1 mm

Table 16.6. Algorithm for automatic platform control

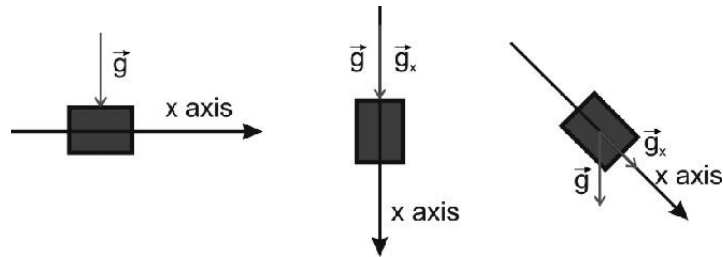


Figure 16.13. Components of gravitational acceleration vector

When the referent axis for the measurement of the angle (X axis in Figure 16.13) is parallel to the Earth's surface, the value of relative gravitational acceleration is 0. When the referent axis is perpendicular to the Earth's surface, relative gravitational acceleration is 1. In any other position, the component of gravitational acceleration on each of the referent axis will have a value between

0 and 1. These values are then measured by the accelerometer, and based on them, tilt angles can be calculated as described in step 3 of the pseudocode shown in Table 16.6. By using simple trigonometric operations, it is possible to determine the sign of the tilt angle of the platform relative to the ground. The value and the sign of the calculated angle is then used in the decision-making process which results in instructions that consist of extraction or retraction of each of the linear actuators. The ± 1 degree dead zone was chosen so the algorithm would not be influenced by small vibrations and smaller tilt angles. The algorithm described is repeated periodically (5 ms period).

16.5. Results

After assembling the robot, a thorough testing of the performance of the overall system was done. Tests were mostly conducted by an human operator, with the robot in the manual mode of operation. During the tests, several important parameters that describe the robot's behavior were observed.

The first parameter observed was the degree of autonomy of the mobile robot, which was determined by measuring the time that it takes the battery to discharge to the level at which the normal operation of the robot is impossible. The robot's battery discharge time lasted from 5 to 7 h of operation, dependant on the load that was on the robot platform. Increased battery drain and current as well as the decrease of the robot's operational time occur when the robot is used in an environment with angular slopes. When the robot is moving up the slope, besides the increased motor current, there is also a significant current being drawn by the linear actuators that are keeping the load of the robot in a horizontal position. These are some identified factors that affect the battery life.

A second parameter was the maximum distance between the operator and the robot on which the uninterrupted control of the robot is still possible. The control loss occurs due to the limited range of the wireless ZigBee transceivers on the joystick and on the robot. Maximum range at which the robot was still operable was about 30 m. This experiment was conducted in an indoor environment with reinforced concrete walls that significantly weakened the signal. If the robot is used in an outdoor environment, the range is increased up to 100 meters. It is possible to achieve these ranges because wireless transceivers with external antennas are used.

The main purpose of the robot is to transport and lift heavy materials and objects, and for that reason it was very important to check the maximum load

that the robot can carry. Although only a prototype built on a small scale, a significant amount of payload can be carried around on the robot's platform, without affecting the performance of the robot. The upper limit for the payload is set at 5 kg, although the robot can carry even more, but with noticeable effects on its performance.

After the experiments in which the behavior of the robot was monitored and observed, the robots capability of executing some simple tasks autonomously was put to the test. To show an autonomous operation of the robot, a simple experiment that includes operation of almost all systems of the robot was designed. The scenario of the experiment is the following:

The robot is placed in a controlled environment ($3\text{ m} \times 3\text{ m}$ polygon), together with the red colored, round-shaped target, glued to one of the walls of the polygon. The task was to have the robot rotate on the spot, until the red target is detected. After detection, the robot should move toward the target and stop at the distance of approximately 20 cm from the wall. In the end, the robot should calculate the distance from its initial position to the place where it stopped moving. To do such an experiment, a laptop computer was placed on the platform and connected to a kinect camera and to the main module of the robot. The image from the kinect camera is used to detect and identify the marker (target), while the ultrasonic range finders were used to determine the distance from the wall. A simple color detection algorithm was implemented on the computer, using Matlab[®] program. Besides color detection, a simple control algorithm, which sends the commands to the main module, was also implemented in Matlab[®]. In this case, the main module was providing the data from the encoders and the ultrasonic range finders to the computer. The computer used these data to determine the distance of the robot from the wall as well as the total covered distance. Some images that were captured during the experiment are shown in Figure 16.14.

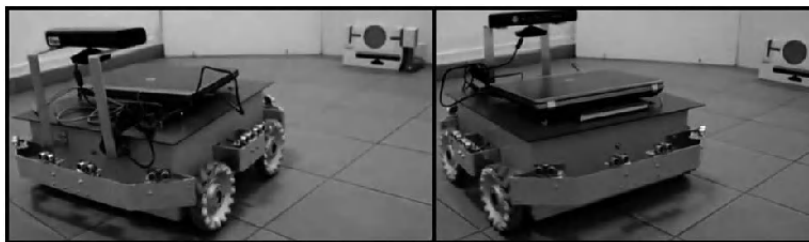


Figure 16.14. *Robot performing tasks autonomously*

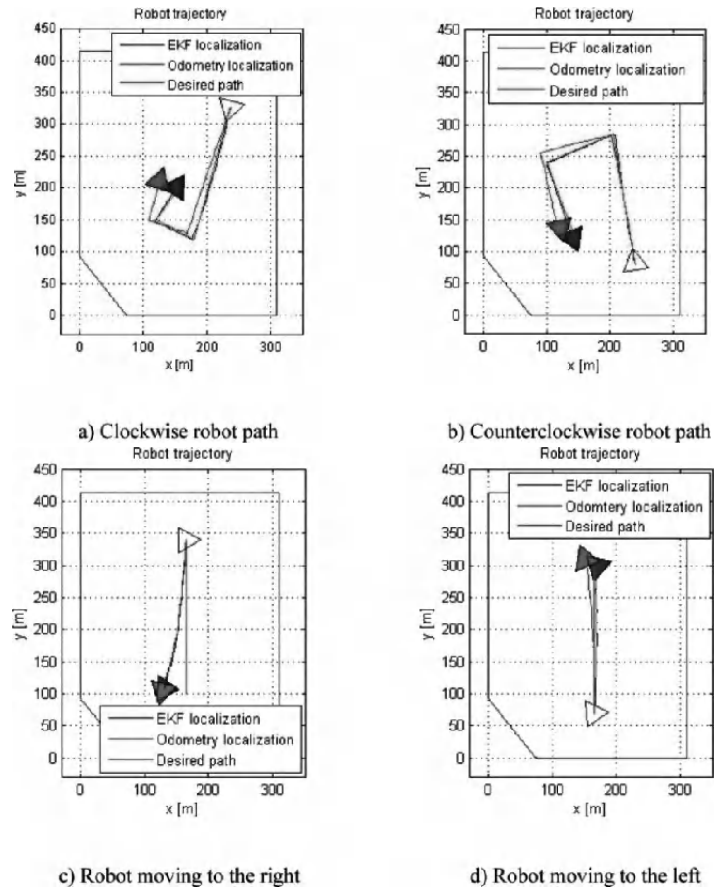


Figure 16.15. Experiments with localization of the robot

With all sensors and systems checked and proved working, it was possible to implement some more complex algorithms. One of the main problems in mobile robotics is localization or determining the position and orientation of mobile robots [HAB 07]. Thus, the localization algorithm was implemented in two ways (odometry- and landmark-based localization using an extended Kalman filter). In total, four experiments with localization were conducted. In every experiment, both the odometry- and landmark-based localization algorithms were running simultaneously. Figure 16.15 shows the results of the experiments after running tests with both algorithms. The first two experiments (Figures 16.15(a) and (b)) refer to the robot moving in the

way in which the non-holonomous robots move, while the other two experiments (Figures 16.15(c) and (d)) emphasize the robot's possibility for holonomous movements. Transparent triangles refer to the start point of the robot, while dark gray and light gray triangles refer to the end point of the robot. The dark gray triangle (and dark gray trajectory) is the result from landmark-based localization, while the light gray triangle (and light gray trajectory) is the result from odometry localization. As expected, the results gained from the experiments with localization differ, and also, the position and orientation of the robot obtained from landmark-based localization with extended Kalman filter (EKF) are more accurate than the results obtained using only the odometry localization. The reasons for such results lie in the difference of the operation of the two algorithms and also in the number of sensors required by the algorithms. A detailed analysis of the localization results is beyond the scope of this chapter.

16.6. Conclusion

The design and building of a mobile robot platform – HOLBOS explained in previous sections – offers a multipurpose lifting and transportation solution incorporating most of the commercially available solutions in this area. Although the robot is built on a small scale, it proves its effectiveness throughout the conducted tests and experiments.

A user-friendly manual control interface makes the robot easy to operate for an average user. Its robustness and the significant weight that the robot can lift and transport, makes it operable for industrial conditions, regardless of its small-scale design. The mecanum wheels allow the omnidirectional movement of the robot that significantly improves its handling in tight areas that are often encountered in industry or on assembly lines. The autonomy given by a long-life battery power supply and wireless control allows the user to manage the transport tasks with ease, intuitively, and without any special training.

When compared to commercially available solutions, few improvements are noticeable:

- Although cheap to build, the Holbos robot represents a complete and functional solution capable of performing multiple different tasks simultaneously, as described in this chapter.
- The autonomous leveling of the platform, while simultaneously carrying the load, expands the borders for possible areas of the robot's use. This makes it capable of working not only on flat, but also on sloped surfaces.

– The use of a kinect camera and ultrasonic sensors, together with the possibility of a computer-based control, gives the possibility of an autonomous, vision-orientated control of the robot. This also makes a good base for the implementation and testing of other control algorithms for the autonomous control of the robot, based on data from the kinect camera or the ultrasonic range finder sensors.

– The wireless, ZigBee-based, control can be used to reprogram the robot's tasks and its paths from a central control computer or system, therefore allowing groups of the same or similar robots to be programmed and controlled at the same time, enhancing their performance and effectiveness. This not only saves money being spent on labor power, but also saves time needed for lifting and the transportation tasks in production. It also makes these tasks less dangerous and easier for humans.

– Holbos is intended to be a platform for further development, not only for the development of algorithms, but also for different tasks that the robot should fulfill, either in industry or in some other areas.

– Based on these improvements, the Holbos robot is already being used as a platform for development and testing of different control algorithms in the Faculty of Electrical Engineering in Sarajevo, therefore becoming the didactic and research and development tool.

16.7. Bibliography

- [AYR 81] AYRES R., MILLE S., *The Impacts of Industrial Robots*, Carnegie-Mellon University, Pittsburgh, PA, 1981.
- [BLA 90] BLACKWELL M., The URANUS mobile robot, Technical Report CMU-R-TR-91-06, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, September 1990.
- [BUR 98] BURGARD W., CREAMERS A.B., FOX D., *et al.*, "The interactive museum tour – guide robot", *Proceeding of the 15th National Conference on Artificial Intelligence*, Madison, WI, 1998.
- [DWI 12] DWIVEDI J., MAHGOUB I., "Robotic surgery – a review on recent advances in surgical robotic systems", *Florida Conference on Recent Advances in Robotics*, Boca Raton, FL, May 2012.
- [FAR 12] FARES K.N., EL-HAGRY M.T., EL-KOUSY A.A., "Mobile robot navigation in warehouses for materials handling", *4th Saudi Technical Conference and Exhibition*, STCEX 2006, vol. 3, pp. 144–150, 2006.

- [FLI 11] FLIPSE M., Altering and improving Kiva, some suggestions for improvement of the current Kiva system, Technical Report, Vrije Universiteit, Artificial Intelligence Department, 2011
- [HAB 07] HABIT M.K., “Real time mapping and dynamic navigation for mobile robots”, *International Journal of Advanced Robotic Systems*, vol. 4, no. 3, pp. 323–338, 2007.
- [KUR 06] KUROI K., ISHIHARA H., “A four-leg locomotion robot for heavy load transportation”, *International Symposium on Automation and Robotics in Construction (ISARC)*, Beijing, 9–15 October 2006.
- [MCM 06] MCMAHAN W., CHITRAKARAN V., CSENSITS M., *et al.*, “Field trials and testing of the OctArm continuum manipulator”, *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, Orlando, FL, May 2006.
- [TLA 08] TLALE N., DE VILLERS M., “Kinematics and dynamics modelling of a mecanum wheeled mobile platform”, *Council for Scientific and Industrial Research*, Pretoria, South Africa, 2008.
- [VEL 12] VELAGIC J., *Mobilna Robotika*, Sarajevo, 2012.
- [WAL 08] WALLIN J., The history of the industrial robot, Technical Report from Automatic Control at Linköpings University, Linköping, Sweden, May 2008.
- [ZAM 11] ZAMBACH S., LOMONACO L., The influence of robots on the human society, Technical Report ID-7219, Roskilde Universitet, Denmark, 2011.

Chapter 17

Advanced Artificial Vision and Mobile Devices for New Applications in Learning, Entertainment and Cultural Heritage Domains

In this chapter, the challenges posed by mechatronics and education [HAB 06] are addressed in the field of advanced artificial vision technologies and mobile devices. In particular, the chapter presents the emerging technologies for mobile learning and introduces two novel research projects. State-of-the-art technologies and applications in mobile learning are presented in section 17.3. In section 17.4, a detailed description of two projects headed by the Artificial Vision and Real Time Systems (AVIRES) laboratory at the University of Udine for mobile learning, entertainment and cultural heritage domains is provided.

17.1. Introduction

In the past few years, mobile devices have been boosted by very strong technological innovations. Mobile devices can be considered as (Web-enabled) devices that are generally used not within a fixed location

Chapter written by Gian Luca FORESTI, Niki MARTINEL, Christian MICHELONI and Marco VERNIER.

but have been conceived and designed to be portable and usable whilst on the move. Typical mobile devices include Web-enabled cell phones (mobile phones) and Web-enabled pocket-sized Personal Digital Assistants (PDAs) [W3C 05]. The first mobile phone appeared in police cars in Detroit in 1921 and in 1946 they had already reached the commercial service level. These devices communicate with each other with a single high-powered antenna installed on a skyscraper that allows a transmission range of about 100 km. Over the years the advent of transistors led to an impetuous leap in technology: the possibility of building smaller and sophisticated electronic components enabled phone companies to develop a large number of models, more usable, comfortable and with other rich features. For example, a smartphone is the term used to specify a new generation of mobile devices that let users have a large number of features and services. As reported in [GAR 11] the world's leading company in information technology research and advisory sold 440.5 million units in the third quarter of 2011 (up 5.6% from the same period of the previous year).

Another important aspect that is strongly connected with the diffusion of mobile devices is the innovation of the communication infrastructure. Nowadays the community is talking about the fourth generation of mobile networks with the goal of reaching a transmission data speed of 1 GB/s. The new, more reliable and faster network infrastructures have pushed more and more the mobile device companies to make new mobile phones that could take advantage of these resources and new services. The technological progress leads to the design and development of new, more sophisticated and innovative, mobile phones. While previously, mobile phones were equipped with a single physical keyboard, they are now equipped with a large liquid crystal display (LCD) screens and fast microprocessors that allow the users an excellent user-experience and the performing of tasks, such as multimedia processing and data logistics. In the past few years, smartphones have exploited the touch screen technologies introduced by PDAs and other handheld devices: the keyboard has been replaced by a screen that allows the user to interact with a graphical interface using fingers or other pointing devices. The PDA can be seen as the first generation of mobile devices/phones. The second generation is now represented by smartphones and tablets. A PDA is a handheld computing device that combines multiple functions and features, including telephone, fax, Internet and networking or other forms of different connectivity capabilities. These devices are mainly used by users who need to compute operations while moving, simply called "Mobile Computing". Before that such features were provided only by laptops or desktop computers. The PDA is the first step in a future trend that would be later defined as "technological convergence".

The next generation of mobile computing, represented at the forefront by smartphones, will foster the convergence of communication, computing and consumer electronics; three traditionally distinct industries with quite low interoperability. While previously, mobile phones were equipped with a simple address book and agenda, now a smartphone has several features, such as a camera, a voice control and so on. So, a smartphone can be seen as a mobile phone with computer capabilities that allows us to interact with computerized systems, send email and access to the Web [DIC].

According to [ZHE 05], the communication and computing environment is moving toward interacting with the physical environment or even becoming part of it. As mentioned by Mark Weiser [WEI 99], chief scientist at Xerox PARC and also considered the father of Ubiquitous Computing, in the 21st Century the technology revolution will move into “the every, the small and the invisible”. That is what is really happening now. This leads to a real-world environment where the information processing moves to the background and the humans concentrate on the tasks, and not on the tool. In this sense, the technology should be considered as a tool to serve the needs of people, not something to depend on. Weiser [WEI 99] suggested that the deepest technologies are those that disappear (“Disappearing Technology”). They weave themselves into the fabric of everyday life until they are indistinguishable from it. The concepts of access anywhere and anytime from any device are intertwined with the progress of the network infrastructures and with the aspects of convergence. Furthermore, smartphones can be considered as a technological expression that offers a complete set of services, including those of a normal cell phone, PDA and wireless communication facilities.

According to Gartner’s Press Releases [GAR 11], smartphone sales reached 115 million units in the third quarter of 2011, up 42% from the third quarter of 2010. Another kind of mobile device that reached a very significant success in the past few years, and whose features are very similar to a smartphone and a computer, is called a tablet. A tablet can be defined as a “flat” notebook uniquely composed of a touch screen where users interact through direct touching on the screen of the device. This new generation of computers is enjoying a great success: according to IDC [IDC 12] the sale of tablets is estimated at around 160 million devices in 2012.

Both smartphones and tablets have quite similar, though at the same time different, operating systems that support the devices just described. As shown in Figure 17.1, Symbian OS, Android OS, Apple iOS and Microsoft Windows Mobile OS are the most diffused operating systems.

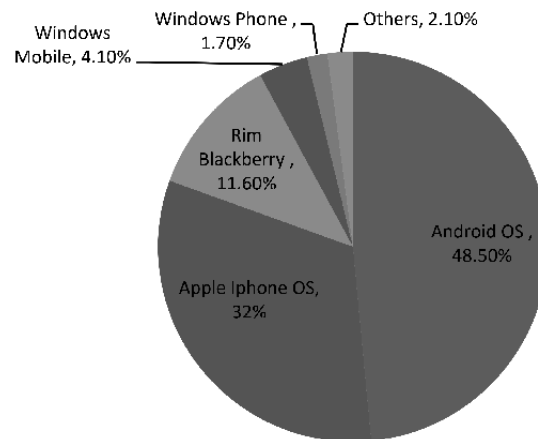


Figure 17.1. Smartphones used by 50.4% of U.S. consumers, 48.5% of them use Android. For a color version of this figure please see www.iste.co.uk/mechatroneng.zip

In 2009 the market leader was Symbian OS that covered the majority of Nokia devices. Symbian OS, heir to the operating system Electronic Piece Of Cheese (EPOC), was initially developed by a joint venture between Psion, Ericsson, Motorola and Nokia. However, in 2008 Nokia decided to buy all the shares of the other companies in order to become the only owner. Later, the transition of Symbian Foundation allowed for making for this operating system open source. Nevertheless, especially in the past two years, the Google Android operating system achieved a huge boost. In 2011 it accounted for 52.5% of smartphone sales to end users in the third quarter of 2011, more than doubling its market share from the third quarter of 2010. Certainly, a historical strategy was the purchase of the Motorola Mobility Company (August 2012), so Google definitively entered into the hardware market. Another company that is fighting for a slice of the market is certainly Apple Inc. In the past years, Apple Inc., led by its chief executive officer Steve Jobs, introduced a new and highly innovative device: the iPhone. It introduced a complete new generation of smartphones. The iPhone was the first mobile device that introduced a revolutionary way to conceive and design the phone in terms of the user's experience. The numerous applications, simply called "apps" and available in the "Apple store", have become the new interface which allows the user to have the maximum benefits using this newfangled mobile phone. Microsoft participates in this mobile device challenge as well. At the last Mobile World Congress, 2010, the company presented the new operating systems called Windows Phone 7. Compared with the previous versions of the

operating systems, the marketing strategy was aimed at a consumer audience rather than an enterprise audience. Nevertheless, this operative system remains at sixth place with a total of 1,701.9 million units sold in the past year. As shown in Table 17.1 there are other operating systems, such as Research In Motion, (BlackBerry - RIM) and Bada (Samsung Electronics), that are placed in the mid range compared with Apple and Android.

Operating system	Units 3Q11	Share 3Q11 (%)	Units 3Q10	Share Q11 (%)
Android	60,490.4	52.5	20,544.0	25.3
Symbian	19,500.1	16.9	229,480.1	36.3
IOS	17,295.3	15.0	13,484.4	16.6
Research In Motion	12,701.1	11.0	12,508.3	15.4
Bada	2,478.5	2.2	920.6	1.1
Microsoft	1,701.9	1.5	2,203.9	2.7
Others	1,018.1	0.9	1,991.3	2.5

Table 17.1. *Main producers of smartphone operating systems*

Several studies have been conducted to verify the smartphones usage by age: according to Nielsen's studies [NIE 12b], during Q2 of 2012, the smartphone market is still growing. This growth is driven by increasing smartphone purchases: two out of three U.S. citizens who acquired a new mobile phone in the last three months chose a smartphone instead of a feature phone. Nielsen [NIE 12a] showed that teenagers between 13 and 17 years old demonstrated the most dramatic increases in smartphone adoption, with the majority of American teens (58%) owning a smartphone, compared to roughly a third (36%) of teens saying they owned a smartphone just a year ago. And they just seem to be getting younger. A study conducted in April 2012 by the Massachusetts Aggression Reduction Center showed that 20% of third grade students have cell phones while 83% of middle school kids have a mobile device. These information are significant considering the new proposals in education and in pedagogy that are spreading in these last years.

17.2. Chapter contributions

Mechatronics is defined as the science that studies how to make different disciplines interact together and it is strongly involved in the new learning

and educational processes [HAB 08]. New generation mobile devices such as smartphones and tablets are new communication systems that introduce novel learning methods. In the past few years, mobile devices together with new educational methodologies have been adopted by schools, museums, etc., giving rise to the new concept of mobile learning. Despite the diffusion of such systems, there are only a few studies that show and study the efficiency and effectiveness of mobile devices for such purposes. In this chapter, the authors contribute to the research in mechatronics and education introducing the most relevant works for mobile learning and education and describing two recent research projects in such areas.

The project entitled “Image Processing Supports HCI in Museum Application” proposes an application to support users learning about paintings. It combines different state-of-the-art computer vision techniques together with augmented reality to display painting information through mobile devices. The main objective was to propose a novel user interface that allows users to interact with the paintings images captured by the mobile device to get information about the paintings themselves (e.g. author, characters and objects represented). Usability studies have been conducted with different classes of users to evaluate the effectiveness, efficiency and satisfaction in use of three different proposed designs.

The project entitled “Back to the future” is a three-dimensional (3D) image gallery developed to increase the knowledge of the cultural heritages and the history of a city. The application has been designed both for desktop and portable devices, such as smartphones and tablets. A 3D environment is presented and the user is invited to explore it as a kind of tridimensional game. 3D images of the most important places of a given city are presented and the user can see them using 3D glasses.

17.3. Mobile devices for education purposes

The diffusion of more and more sophisticated mobile devices, the innovations of the communication infrastructures and the new ways to use these devices lead to new aspects in education and pedagogy. Several studies investigate how these new technologies can support a new way to learn. A new very important concept is now widespread: mobile learning. It can be defined as any educational provision where the sole or dominant technologies are handheld or palmtop devices [TRA 05]. According to Roschelle [ROS 03], the attention should be focused toward the identification of simple things that technology does extremely and uniquely well, and to understand the

social practices by which those new affordances become powerful educational interventions. Traxler [TRA 05] showed that mobile devices are now radically transforming social notions of discourse and knowledge, and are responsible for new forms of art, employment, language, commerce, deprivation and crime, as well as learning. With increased popular access to information and knowledge anywhere, anytime, the role of education, perhaps especially formal education, is challenged and the relationships between education, society and technology are now more dynamic than ever.

The new concept of m-learning can support a wide range of subjects from childrens education to higher education. For that it is important to know the most significant projects and research that have laid the basis for future studies. Mobile learning introduces several challenging aspects with respect to the traditional learning methods. So it is important to continue investigating its efficiency and effectiveness in education and pedagogy. In the following section, some of the most important and interesting projects that exploit mobile devices for education and pedagogy are described.

17.3.1. *Mobile learning for education*

17.3.1.1. *HandLer*

HandLeR was one of the first projects constructed by the University of Birmingham [SHA 00]. It was started in 1998, and was based on the theory of “learning conversation” with the characteristic to understand how people learn in multiple contexts over their lifetimes. The learning conversation was implemented using different scenarios that included a young boy of 11 years old on a school trip, a radiologist in his/her first academic year in neuroradiology and an elderly person who is rethinking and reorganizing the memories of a lifetime. The project was centered on the implementation of different interactive and portable personal computer contexts with the same features of a tablet (see Figure 17.2).

The main goal of the project was to implement different user contexts. For example, the software for the HandLeR school trip was developed through questionnaires and interviews with children of 11 and 12 years old, and the goal was to create an interactive style connected to the way of how the children learn in the field rather than using the same metaphor of the office desktop. It introduced a specific interface based on the concept of an animated mentor with the aim of helping the children and allowing them to easily learn and interact with it and with the new proposed contents. These experiments were limited by the technology available at that time, although it was a very

important project as it stated the basis for other and future projects with similar goals. These experiments have underlined how portable are these types of devices. They showed that such devices allow the user to learn everywhere, one-to-one, with the capabilities to adapt to the knowledge and the learning style of the one who is learning: non-invasive, always available, adaptable to different contexts, persistent, useful and easy to use.



Figure 17.2. *HandLeR's interfaces example*

17.3.1.2. *M-learning*

M-learning was another important project in mobile learning lead by the English Skill Development Agency (LSDA) and other universities in the UK, Italy and Sweden. This project was performed to support students between 16 and 24 years old, with the characteristic to have lost his/her interest in learning and with unsatisfactory results in the educational system. A learning management system and the interface of a micro-portal were developed with the aim to provide access to the materials and the learning services through several devices, such as mobile devices, Web and TV. Moreover, an SMS system based on health arguments was also developed, with the goal to inform the users on different arguments as, for example, the dangers of drugs, a drive quiz and so on. The project results were satisfactory because it showed that mobile learning is an efficient tool that allows the users to reach places that other forms of learnability are unable to reach. Furthermore, these experiments have shown how it is better to provide different pieces of learning, rather than to offer the user a unique solution.

17.3.1.3. *MOBILearn*

MOBILearn was an European project which involved 24 industrial and academic partners from 10 different countries. The project started in January 2002 and ended in March 2005. The goals were to develop and evaluate a

mobile learning architecture based on effective learning and teaching theories, which were based on a mobile system outside the scholastic environment. The objectives of this project were very wide: the main goal was to allow the users access to the knowledge in an ubiquitous way using innovative mobile interfaces. The project also introduced new learning models for mobile environments, new system architecture to support the making, the supply and the tracking of learning contents, new methodologies to adapt the content for mobile devices, such as smartphones and tablets, and finally new business models with the aim of providing a sustainable deployment of new technologies for mobile learning. The MOBILearn technology was tested in different contexts, for example, in a museum, on a university campus and for first aid.

17.3.1.4. From e-learning to m-learning and mobile learning

From e-learning to m-learning and mobile learning are the projects funded by the European Commission on the through “Leonardo Da Vinci” program and directed by Ericsson [SAM 06]. They aim to support education and professional training using mobile phones and other mobile devices. The major part of the users tested were students. The final evaluation questionnaire showed how they considered positively and funny mobile learning using mobile devices, although only 45% of participants improved with the quality of e-learning. The major problems encountered by the users were the frustration caused by connection problems (continued disconnections) and limitations on visualization due to the small size of screens [NIX 05]. Moreover, from e-learning to m-learning projects solved problems connected with course presentations on a PDA by using the Microsoft Reader Works application.

17.3.1.5. Nintendogs

Nintendogs is a learning project based on a game. It exploits the Nintendo DS Console as a context to create a set of several activities that involves two different classes of students. The main character of the game is a dog for which the students must take care of. The game also involves other activities as, for example, bringing the dog to canine exhibitions with the goal to winning money as the score. It is worth noticing how such activities lead the students to perform activities that are generally considered as boring. The students perform such activities to reach their objectives and collaborate with different students in different classes. These activities have the goal of stimulating the social interactions among students of different ages.

17.3.2. Mobile learning for museums and galleries

Mystery at the Museum is another project described by Cabrera *et al.* in [STO 05]. They describe the experience of designing a collaborative

learning activity for a traditional historical/cultural museum involving small groups of students through mobile handheld devices. Any group is partially informed of the task that they must complete and it receives additional information through a mobile device. For instance, given different brainteasers, the students have to share their answers among all the members of the groups to achieve the final goal. MyArtSpace [VAV 09] allows children, who are visiting a museum with their classmates, to conduct a research connected to the museum. Before visiting the museum, the teacher informs the students about the rules and the goals of the “game” and asks them a main question that students must answer after or during the visit. In this case, the mobile device is represented by a mobile phone used by a student to collect photos, audios, videos and other notes. The data collected are also used during classes to create and share image and video galleries with other students. Gidder [PIE 08] is another interesting project developed for museum purposes. The main project goal is to support collective knowledge within a class and in the museum. Before visiting the museum, students are required to work in groups using the Wiki system. The objective is to prepare some notes about the most interesting paintings they found and to discuss about the exhibition. During the fruition of the paintings session inside the museum, students share and send information to the wiki system and post it on the blog. The collected data are then discussed by all participants during classes. Finally, City [GAL] is a system that allows three users to communicate and, visit at the same time, the Charles Rennie Mackintosh room at the Lighthouse Centre for Architecture, Design and the City in Glasgow. The visitor uses a PDA to interact with the environment and with the capabilities to share with other users his/her position on the map of the gallery. The other two remote users, using a Web application or a virtual reality environment, can share information with users inside the gallery and communicate in real time using an audio channel.

17.3.3. Mobile learning for workplace, professional development and training

It is worth introducing some projects that are not strictly connected with mobile learning in education but which concern work places, professional development and personal training. MeduMobile [SCH 06] uses the video communication technology available on mobile devices and notebooks to support medical students and teachers. The main purpose was to train new students in different contexts as, for example, to improve the relationships between medical staff and patients. These activities were filmed and transmitted in real time to the students at the Charite’ campus.

Flex-Learn [GJE] is a project of the Danish University of Education that aims to support drivers of heavy/long vehicles using different multimedia displayed on mobile devices. These multimedia involve videos, audios, text overlays, etc. The main characteristics of this platform is that users can interact with it, making notes and highlighting favorite content. Know Mobile [SMØ 03] was one of the first projects in the medical research field that aimed to explore how to use mobile technologies and support strategies of Problem-Based Medicine (PBL) and Evidence-Based Medicine (EBM) after the reform of the medical educational system in Norway. One of the main research goals was to investigate the field of the “just in time” in a medical concept, to support medical personnel who must always receive the most updated information (e.g. during an intervention). Chittaro *et al.* [CHI 11] proposed an adaptive Web-based information system called Presydium (Personalized Emergency System for Disabled Humans) that provides tailored instructions in the field of helping medical first responders in dealing with disabled persons. In [BUT 10] the authors developed an adaptive fitness game for mobile devices that takes into account users’ heart rate, age, fitness level, etc., and supports them in exercise at the optimal intensity to gain cardiovascular benefits. This new training system introduced a new way to learn through a mobile device so that the user is completely informed about his/her health status and the best exercise to maintain a good fitness level.

17.4. Image processing supports HCI in museum application

In this section, a novel information visualization technique for mobile devices is presented. The objective of the developed application was to enhance museum paintings consumption displaying additional information to the images acquired by the device’s camera [FOR 09] through augmented reality. In particular, the main goal is to display additional information on the paintings’ characters images acquired by the mobile device. The proposed information visualization technique exploits the most recent algorithms developed in the fields of image processing and computer vision [FOR 05] to introduce a novel way to display augmented reality contents on mobile devices. In particular, the case study concerned usability evaluation of such techniques designed for a traditional, historical and cultural museum. The system introduces a novel method to learn and educate people about art, in a multimedia museum [FIN 12], through mobile devices. At the time of writing this book there are no similar works where the end user is supported by an augmented reality technique to visualize information related to a painting’s characters. In [VLA 01] and [DAH 02] the mobile “archeoguide” application has been introduced to reconstruct historical sites. Takacs *et al.* [TAK 08]

proposed an outdoor augmented reality system for mobile phones using Loxel-based visual feature organization. Despite this, there are several studies that show how to correctly visualize information on mobile devices. In [CAR 07] and [BUR 07] two different approaches for geographical map visualization on mobile devices are introduced. Several designs have also been proposed for Personal Information Management (PIM) devices [ROB 08].

Usability evaluation of information visualization techniques for mobile devices is a fairly recent research area. Despite desktop software products some base rules have been defined to achieve good usability results; there are no clearly defined methods that could also be applied to mobile devices interface designs. This is especially true for augmented reality applications, since only the recent developments in sensor [PIC 09] and device capabilities have allowed us to process and display contents in real time. The task to achieve the given goal was hard, since there are no strict and well-defined methodologies that could be exploited to conduct usability evaluation studies for such systems. As pointed out in [STO 05], there are three fundamental difficulties reported in the literature [KJE 04] regarding usability studies: (1) it is hard to provide realistic studies that capture all the relevant situations; (2) it is difficult to apply standard evaluation methods such as observation and thinking aloud; and (3) the data collection procedure is very complicated and the control over the environment is very limited. Laboratory evaluation studies help to deal with such problems, but at the same time they introduce a lack of realism. In the literature several approaches have been proposed to recreate or simulate the real context of use. Most of the times the proposed solutions are almost impossible to achieve since it is very hard to realistically recreate the context and the surroundings within restricted laboratory spaces. Usability evaluations have been performed in laboratory-based settings by simulating activities of typical users with a considerable degree of realism. To reach the proposed goal and to evaluate the performance of the proposed system with respect to a simulated–real scenario, a dataset of paintings has been built by taking pictures of paintings available from the Internet. The same paintings have been printed out and arranged within the laboratory space to evaluate the performance of the system within a simulated–real scenario. To achieve the best usability performances three different designs have been proposed and evaluated by means of standard evaluation methods.

To achieve the proposed objective three main goals have been reached: (1) recognize each single painting within the museum dataset, (2) find the transformation function between the real world and the device world and (3) display augmented reality information in a way that would be useful and unobtrusive. Each of them has been addressed by a single module of the entire system that will be described in the following sections.

17.4.1. *System description*

The objective of the proposed technique is to enhance museum paintings consumption by displaying additional information to the current acquired image by means of augmented reality. The overall system can be grouped into two main phases: (1) off-line signatures computation; (2) online signature computation, matching and information visualization. Image processing and computer vision algorithms [FOR 05] are exploited by the first module of the entire system to output valuable information that could be used to design the information visualization technique. Thus, the three image processing and computer vision system's modules are defined by: (1) the boundary detector module, (2) the signature computation and signature matching module and (3) the homography computation module. Only when the homography transformation from the initial world coordinate system to the current device coordinate system is fairly estimated, can the Human Device Interface (HDI) module be used to provide augmented reality information to the end user.

The boundary detector module extracts only the relevant painted region of a given image and aligns the image boundaries such that they are orthogonal to each other. The signature computation module extracts two different visual-based features from such relevant regions to compute a discriminative signature [MAR 12]. Then, the computed signatures are stored in a dataset that is used for comparison with real-time acquired images. These two steps mainly define the off-line phase. The online phase exploits the same two modules and, in addition to them, the signature matching and the HDI modules are used to achieve the given goal. Given a valid signatures match computed by the signature matching module, the homography computation module is in charge to estimate the homography transformation between the original image and the acquired image. The estimated homography transformation is finally exploited by the HDI module to introduce the novel information visualization technique.

17.4.2. *Boundary detector*

The boundary detector module has to extract only the relevant painting area from a given image acquired by the device sensors. It exploits the Hough Transform (HT) to achieve the goal. Each frame F processed by the camera is initially converted from the RGB color space to the standard eight bit grayscale. The transformed image F^G is then convoluted with a Sobel filter to get the edges image F^E used to compute the HT accumulator space. The HT technique parametrizes lines by considering points as a function of an angle

normal to the line passing through the origin of the image. Thus, given any point (x, y) of the image F^G , its polar coordinates transformation is computed as:

$$\rho = x \cos(\theta) + y \sin(\theta) \quad [17.1]$$

where ρ is the length of the line normal to the line passing (x, y) , and θ is the angle of the line normal. The HT accumulator space F^A can be computed by counting the ρ and θ values for all the points (x, y) . Given the computed accumulator space F^A , the goal is to extract only the relevant painting image region F^R . This is achieved by computing the smallest area – enclosed by the detected lines intersections – of the image that is at least bigger than s times the whole image area. Even this is not guaranteed to give perfect results – since multiple lines can be detected or the whole painting area is not visible – this is enough for the given purposes.

17.4.3. Signature computation

Given the relevant frame region F^R , the signature computation module has the goal to extract visual-based features to compute a discriminative signature. The signature computation and matching module can be grouped into two submodules: (1) the image transformation and feature extraction and (2) the signature computation.

Given the relevant painting region F^R , two kinds of features are extracted: (1) Speeded-Up Robust Features (SURF) and (2) Pyramid Histogram of Oriented Gradients (PHOG). Before extracting such features F^R is transformed as:

$$F^{R'} = T F^R \quad [17.2]$$

such that the detected region boundaries are orthogonal to each other. The given region $F^{R'}$ is also projected to the HSV (hue, saturation, value) color space to achieve illumination and color invariance.

SURF features allow us to detect interesting points in a given image that are highly discriminative and robust to several issues, such as image rotations and scaling. SURF features are extracted from the given painting region $F^{R'}$ by exploiting the standard integral image representation that allows us to achieve lower computational costs. After detecting the interest points $F_{S_p}^{R'}$, the SURF feature descriptors $F_{S_d}^{R'}$ are computed as the sum of the Haar wavelet response

around each of them. Then the PHOG feature is computed. The PHOG feature exploits the histogram of the oriented gradient descriptor and the spatial image pyramid representation to capture information about the shape and the whole appearance of the given region $F^{R'}$. A PHOG matrix $F_P^{R'} \in \mathbb{R}^{m \times 3}$ is extracted by concatenating the PHOG histograms extracted from the three image channels. m represents the total number of histogram bins computed by exploiting the original weighted combination of histograms extracted at the different levels of the spatial pyramid representation. Thus, given a frame F , the signature $\hat{F}$ is defined as $\hat{F} = \langle F_S^{R'}, F_P^{R'} \rangle$. During the off-line phase each computed signature is stored in a dataset that is used for comparison during the online phase.

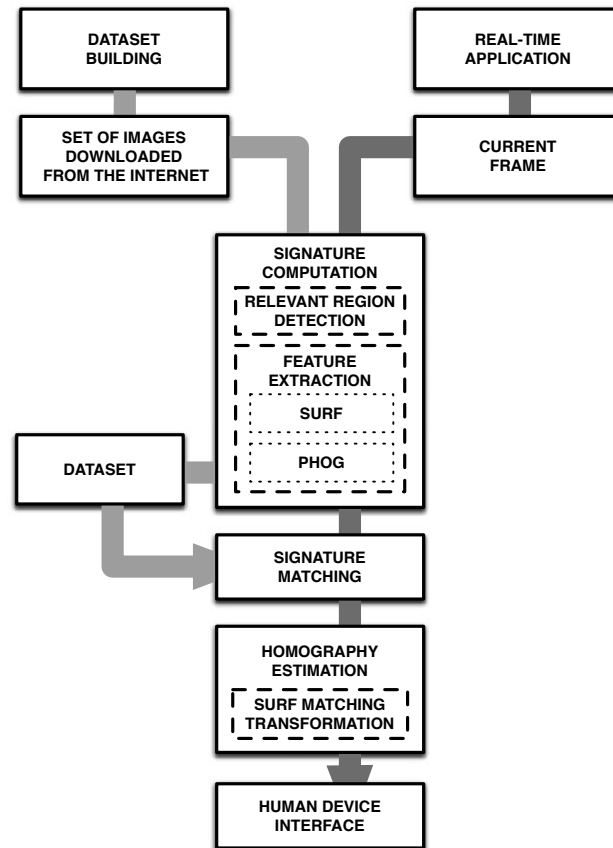


Figure 17.3. Proposed system

17.4.4. Signature matching

During the online phase the signature matching module is exploited to identify if there exists a match between the query signature, say $\hat{Q}$, with one of the previously computed signatures that are stored in the paintings dataset. Let $\hat{Q}$ and $\hat{I}$ be the paintings signatures of the query painting Q and the dataset painting I , respectively. To identify a valid match between the two signatures a linear combination of two distance measures computed between the signatures features is exploited.

Let i, j be a match between two SURF feature descriptors, such that the distance:

$$d_{l2}(Q_{S_{d,i}}^{R'}, I_{S_{d,j}}^{R'}) = \|Q_{S_{d,i}}^{R'} - I_{S_{d,j}}^{R'}\|_2 \quad [17.3]$$

is lower than a fixed threshold Th . $Q_{S_{d,i}}^{R'}$ and $I_{S_{d,j}}^{R'}$ are the i th and the j th SURF features descriptors of the paintings' signatures SURF features $\hat{Q}$ and $\hat{I}$. The overall SURF features distance is computed as:

$$d_{SURF}(Q_S^{R'}, I_S^{R'}) = \frac{1 + \sum_{i,j \in matches} d_{l2}(Q_{S_{d,i}}^{R'}, I_{S_{d,j}}^{R'})}{count(matches) + \epsilon} \quad [17.4]$$

where *matches* are the matching SURF features descriptors.

PHOG features distance is computed by exploiting a linear weighted χ^2 distance as proposed in [MAR 12]. Given the PHOG feature matrices of two signatures $Q_P^{R'}$ and $I_P^{R'}$, the PHOG distance is computed as:

$$d_{PHOG}(Q_P^{R'}, I_P^{R'}) = \sum_c \lambda_c \chi^2(Q_{P,c}^{R'}, I_{P,c}^{R'}) \quad [17.5]$$

where $Q_{P,c}^{R'}$ and $I_{P,c}^{R'}$ are the PHOG feature vectors computed for the two signatures on image channel c . λ_c is the normalization weight.

Finally, a query signature $\hat{Q}$ and a dataset signature $\hat{I}$ are classified [MIC 12] by computing the minimum $d(\hat{Q}, \hat{I})$ distance, such that:

$$d(\hat{Q}, \hat{I}) = \alpha d_{SURF}(Q_S^{R'}, I_S^{R'}) + \beta d_{PHOG}(Q_P^{R'}, I_P^{R'}) \quad [17.6]$$

where α and β are the normalization weights.

17.4.5. Homography estimation

The homography estimation module is used to compute the homography transformation matrix that allows us to display a painting characters' additional information to the device display. To achieve such a goal the module exploits the feature matching technique proposed in [BRO 06]. Since the initial transformation T is applied to region F^R , its boundaries are orthogonal to each other and, as for small changes in image position, the feature transformation is affine, SURF features have been exploited to reach the task.

Let Q and I be two matching painting images such that $I = \arg \min_I d(\hat{Q}, \hat{I})$, where Q and I are the query painting and the dataset painting, respectively. Given all the matches q, i between SURF feature descriptors (see section 17.4.4), the goal is to estimate the homography transformation $H^{Q,I}$ such that $Q_{S_p}^{R'} = H^{Q,I} I_{S_p}^{R'}$. Since the technique proposed in [BRO 06] has been exploited, the readers are asked to refer to [BRO 06] for further details about the estimation of $H^{Q,I}$. Given the matrices $H^{Q,I}$ and T , it is possible to compute the inverse transformation such that any content given in the original coordinate system can be displayed and aligned with respect to the current acquired image.

17.4.6. Human device interface

Given the homography transformations that allow us to overlap content to the currently acquired image, the objective is to design a novel information visualization technique to allow end users to reach the content/information related to a character in a painting by interacting with the augmented reality displayed information. Standard human-computer interaction methods have been exploited such that the end users could find the correct way to interact with the user interface and access the information without any cognitive effort. To reach such a goal three designs have been proposed and implemented respecting the so-called usability "golden rules".

Three different user interfaces have been designed and evaluated exploiting the standard usability evaluation methods that allowed us to identify the best design among them. The three proposed user interfaces have been designed as follows (see also Figure 17.6):

- 1) Painting characters' edges are highlighted with the same color and their names are shown close to themselves. The end user can access character information by selecting the displayed label.

2) Painting characters' edges are highlighted with the same color as before, but name labels are replaced by blinking white circles. The end user has to select the circle to access the character information.

3) Characters' edges are displayed with different colors, and characters' silhouettes are overlapped with semi-transparent colored and blinking silhouettes. The end users have to select the semi-transparent colored silhouette to access the character information.

The proposed information visualization techniques have been designed according to the software engineering process. Such iterative and trial-and-error processes takes place between the designers and the end users. The main advantages of using it are: (1) it is available from the early stages as a low-functionality version of the product; (2) each prototype can be tested with end users at every stage; and (3) it allows us to understand if requirements are correct and if the design and development process is on the right path, then the risk of designing an unusable product is minimized.

17.4.7. Experimental results

According to [CHE 00] usability testing and controlled experiments are the backbone of prototype evaluation. To evaluate the proposed designs two types of tests have been performed: (1) inspection tests (usability inspection is the generic name for a set of cost-effective ways of evaluating user interfaces to find usability problems [NIE 94]) and (2) end user tests. Inspection tests have been performed by usability experts without the direct involvement of the end users. Such a technique is used throughout the overall design process.

Two types of inspection tests have been exploited: (1) *heuristics* tests, that is analytical evaluation techniques that provide opinions, and (2) *cognitive walkthrough* tests, where the HCI experts examine the elementary actions that end user needs to take to achieve a given goal. Evaluators aim to reconstruct the mental processes that end users build during their tasks.

According to [BOR 97], working with end users is a very important task in usability studies. As described in [NIE 93] to reveal about 80% of all usability problems just five testers are required. Faulkner [FAU 03] shows that five participants are not sufficient to detect all the usability issues while 60 of them – randomly selected as sets of five – can detect 99% of the issues.

To evaluate the proposed system 10 users have been tested for any interface with a total of 30 users (see Figure 17.4). To increase the end users interests

they were paid for each session test. Relevant end users' skills are shown in Figure 17.5.

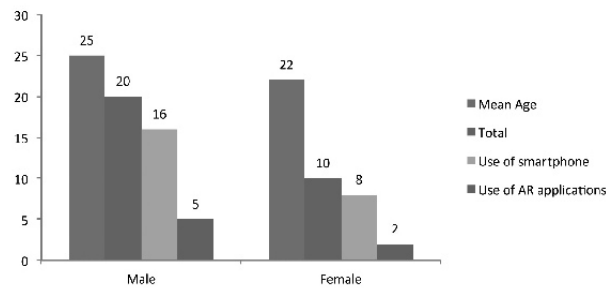


Figure 17.4. Testers' profiles. A total of 30 testers have been selected to perform the required task. For a color version of this figure please see www.iste.co.uk/mechatroneng.zip

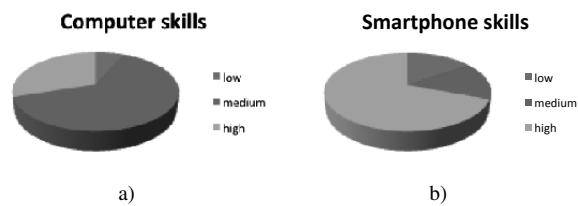


Figure 17.5. End users relevant skills: a) computer skills; b) smartphone skills. For a color version of this figure please see www.iste.co.uk/mechatroneng.zip

According to [BEV 94], to evaluate the usability performance it normally requires providing at least one measure for each of the quality of use: (1) effectiveness, (2) efficiency and (3) satisfaction. Effectiveness is defined as the measure that relates the goals of using the system to the accuracy and completeness with which these goals can be achieved. It was measured as the number of errors (i.e. wrong touches) occurred during the tasks. The efficiency relates the effectiveness achieved to the expenditure of resources. It was measured as the time spent by the user to complete the task. Finally, the users satisfaction is defined as the state of gratification achieved at the end of (or during) the interaction with the system. It was measured by considering both the comments (negative or not) provided during the task and the final debriefing questionnaire that reported open and Likert scale questions. Session tests were performed in a laboratory where a 2×1.5 m painting is

shown. A four inch (display) Samsung smartphone device with Android OS has been used during evaluation tests. The data of the tests were recorded using a video camera placed behind the user. During the briefing, participants were informed about the purposes of the test, the task and its duration. The users were also asked to fillout a screening questionnaire so that information about users' profiles can be obtained. The "think-aloud" technique has been exploited during test sessions, each of which lasts about 15 min. During the tests the end users were asked to perform a task and to verbalize whatever came into their mind during the performance. After each test a debriefing is exploited to investigate unusual or interesting events that occurred during the session. Statistically significant differences in time, error rate or satisfaction are obviously a plus, but observations recorded during the tests became the basis for refinement or redesign, leading to better implementations, guidelines for designers and refinement of theories.

The three proposed information visualization techniques are shown in Figure 17.6. The first user interface (see Figure 17.6(a)) has been deeply evaluated with the support of end user testings. Six participant out of 10 completed the given task with an average execution time of 8 min 33 secs. As shown in Figure 17.7(a), out of the four users that failed the test, 25% of them selected different areas other than the selectable label; 25% selected the menu button; and the remaining 50% did not achieve the given task at all. After debriefing, 90% of the participants were satisfied about the application but 40% of them stated that the user interface was not clear and cognitively expensive.

The second designed user interface is shown in Figure 17.6(b). After experimental test sessions, only one tester out of 10 failed the test selecting the menu button. A total of 90% of the testers stated that the proposed user interface was clear and was easy to reach the information related to a character. One single tester suggested to display the white circles with different colors. As shown in Figure 17.7 the second designed user interface achieves the best performance both in terms of success rate and average execution times. The average execution time required to achieve the given goal is about 4.1 min.

Finally, the third designed user interface (shown in Figure 17.6(c)) achieved the worst results. Only one tester successfully completed the given task. According to a debriefing questionnaire inspection, 70% of the participants stated that the interface was unclear and 80% of them had difficulties in recognizing the silhouette as a selectable element. Almost all the testers said that the character recognition was difficult due to the overlapping of the colored silhouette.

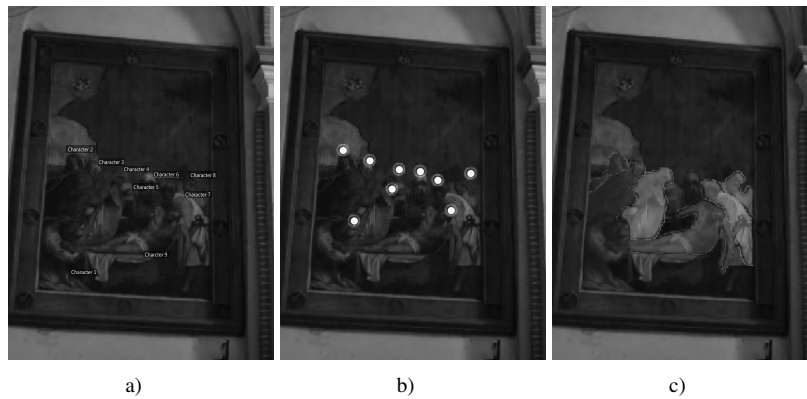


Figure 17.6. *Proposed information visualization techniques: a) painting character's edges are highlighted using colors. Characters' names are used as selectable elements to access character information; b) painting characters' edges are highlighted using colours as in design 1. Characters' names are replaced by blinking white circles that are used as selectable elements to access character information; c) painting characters' edges and silhouettes are highlighted using different colors. Characters' silhouettes are used as selectable elements to access character information*

Figure 17.7 shows that many of the testers that did not achieve the given goal stopped in the grip of frustration and panicked, while just a small percentage of them had randomly touched the screen or selected the menu button.

17.5. Back to the Future: a 3D image gallery

Back to the Future is a project that aims to propose an innovative and futuristic interface to visualize old historical images elaborated and transformed to be displayed in 3D. The developed system has been adapted and tested for different platforms, such as PCs, notebooks, smartphones and tablets supported by Fash Player technology. The system has been mainly developed using HTML 5, Adobe Flash and Papervision. Adobe Flash is a graphical software that allows us to create vector animations on a timeline and it is supported by the scripting language called Adobe Actionscript (AS), which allows developers to create more complicated dynamical applications. Among other characteristics of Adobe Flash, one of the more interesting is the possibility to connect the entire environment with external libraries. To display the 3D images, a dynamical and interactive environment has been created using Papervision 3D. Papervision 3D is simply an extension library that allows us to create a 3D scene inside the flash movie.

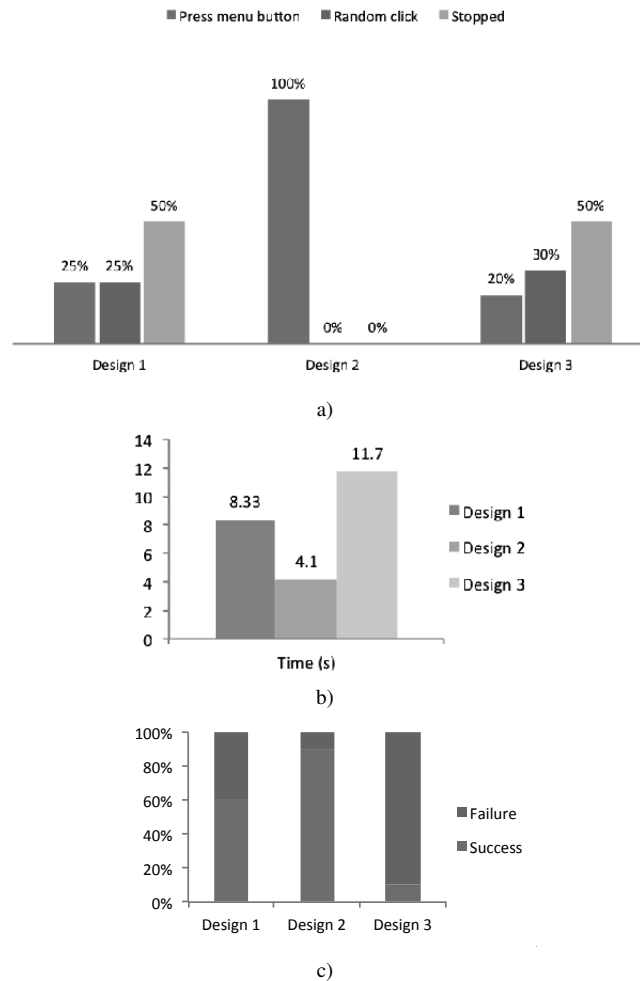


Figure 17.7. Evaluation results: a) most relevant user interface issues: menu button selection, random click on non-selectable area and stopping to reach the given goal; b) average execution time; c) Success and failures. For a color version of this figure please see www.iste.co.uk/mechatroneng.zip

17.5.1. System description

The *Back to the Future* application, as a flowchart is represented in Figure 17.8, starts with a brief animated introduction, played along with a soundtrack of music; the music can be turned off or on by the user using the respective links on the screen interface (see Figure 17.9(a). In the following, a time machine is presented, and the user is invited to enter into the 3D environment by selecting a linked image represented by a gleaming diamond (see Figure 17.9(b).

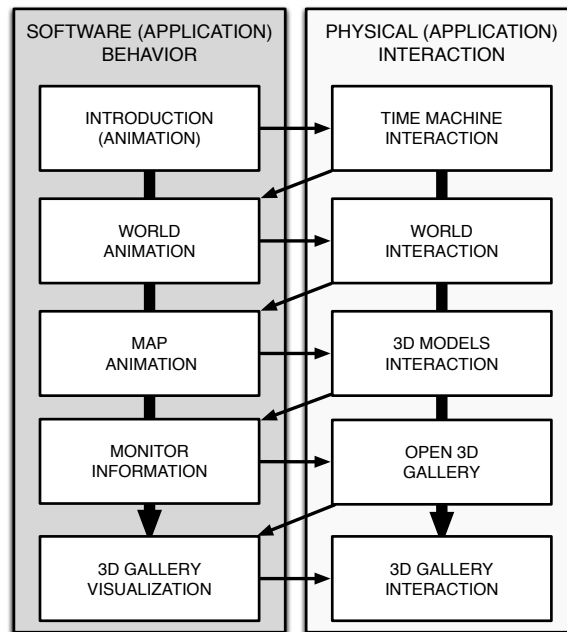


Figure 17.8. System interaction flowchart

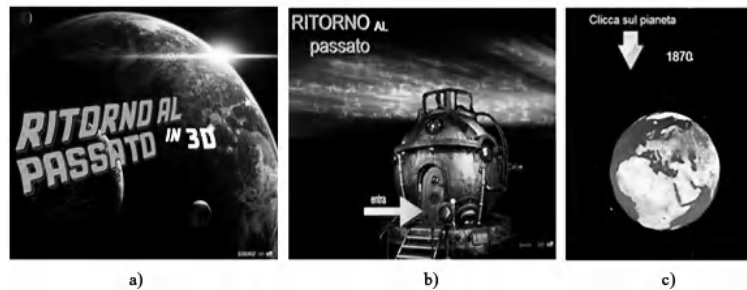


Figure 17.9. a) *Back to the future* introduction (from the Italian version); b) future machine; c) globe rotation

After the user clicked the red diamond, an animation is shown and a 3D globe (Figure 17.9(c)) starts rotating. Then an interactive map of Italy is presented (Figure 17.10(a)). The map is shown in a vertical position and after clicking on it, the map moves to a horizontal position presenting the 3D clickable miniatures of two of the main important Italian monuments: Rome's Coliseum and Venice's St Mark's square (Figure 17.10(b)).

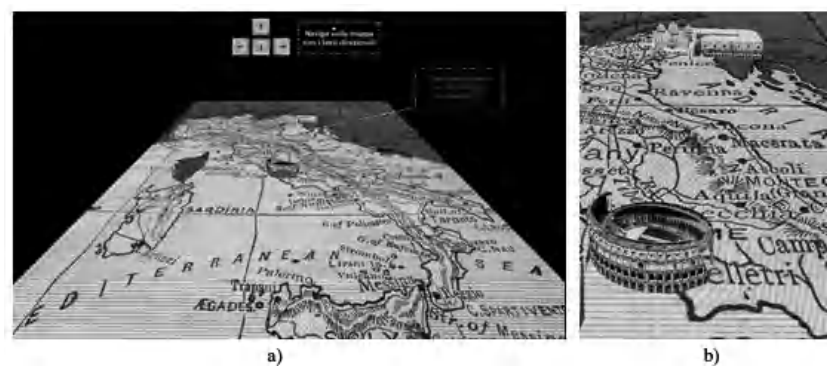


Figure 17.10. a) Interactive map of Italy; b) 3D models of the monuments

The map can be moved using the buttons at the top or by scrolling the mouse if the user is using a normal PC. Both the map and the monuments are 3D reconstructions created as follows: (1) a 3D shape (e.g. a rectangle) has been created using Papervision, then (2) a *.dae* model has been pasted. “*.dae*” models are 3D images downloadable from Google’s archives. If the user clicks or touches the models, the map moves to the bottom of the window and a miniature image and a description of the monument is presented (see Figure 17.11). Then, if the user clicks on the miniature, a carousel gallery of 3D images is shown (see Figure 17.12). The user can move the images by simply moving the mouse or cursor on the left or on the right, or touching them if the user is using a tablet or a smartphone. The images can be viewed if the user uses the right 3D glasses for stereoscopy images. The user can go back to the 3D representation of the map by simply clicking on the exit button positioned on the top of the screen’s interface.



Figure 17.11. *Display information about the selected content*

17.5.2. Testing and evaluations

To evaluate the effectiveness, efficiency and satisfaction of the proposed system, usability tests have been performed. Specifically, user testing has been exploited: they are usability tests entirely based on the evaluation of the system by real potential users of the system. As described in [NIE 93] just five testers are required to reveal about 80% of all usability problems. In the proposed system, a total of 15 users composed of eight males and seven females aged between 20 and 30 years old have been tested. Participants have been divided into three groups: (1) the first group evaluated the application on a desktop PC, (2) the second group evaluated the application on a 10.1 inches Android OS Asus tablet and (3) the third group exploited a four inches Samsung smartphone device with Android OS. Before testing, the users were informed about the goals of the test with some general instructions both on the application and the requirements to complete the test. They were asked to fillout a screening questionnaire so that information about participants profiles can be obtained. The think-aloud protocol was also used during the test sessions. At the end of the test, another questionnaire was proposed to the user to investigate the possible applications' issues. As reported (shown in Figure 17.13) 80% of the participants were satisfied with the system without having found particular problems. The remaining 20% found problems mainly connected to how to move the carousel of images and how to come back to the image description. In the first group, no participants have encountered usability problems. In the second group, only one user had a problem about

how to come back to the description of the image once he/she entered the images carousel. In the third group, a user found problems on how to move the images of the carousel. Finally, among the comments section of the questionnaire we have received some useful information to improve the parts of the application that gave unsatisfactory results to the user. For example, three users out of 15 suggested improving the visibility of the link button (the gleaming diamond) to continue the animation maybe with a larger button or introducing a blinking effect. Lastly, another participant suggested highlighting the 3D images when the mouse is over them.



Figure 17.12. 3D gallery of images of the selected object

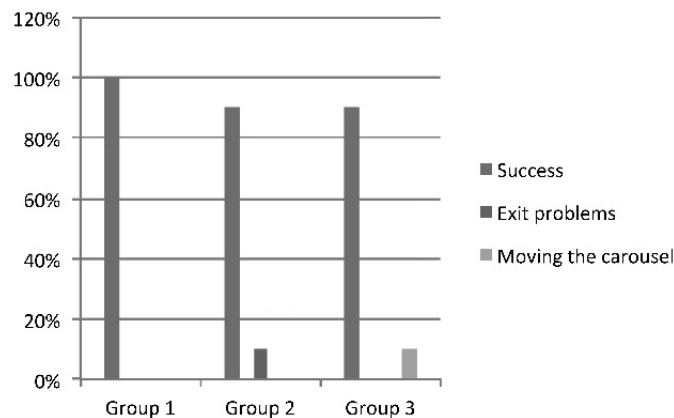


Figure 17.13. Evaluation of the system by the users

17.6. Conclusions and future works

Over the past years, mobile devices have been boosted by relevant technological innovations. Mobile devices have been conceived and designed to be portable and usable while on the move, and recent communication technologies have allowed them to download content from the network every time. The diffusion of sophisticated mobile devices, the innovations of the communication infrastructures and the ways these devices are exploited lead to new aspects in education and pedagogy. Several of these studies have been introduced in the chapter emphasizing the most important concept that is now widespread: mobile learning. Mobile learning is a very interesting emerging topic in the mechanical and electrical research fields. The concept of mobile learning supports a wide range of topics and introduces several challenging aspects to the traditional learning methods. Different challenges related to the mobile learning concepts have been addressed by two proposed works. The works aim to improve the knowledge about paintings and historical buildings through mobile learning. The main objective of both applications is to ease the consumption of contents displayed through mobile devices. The results achieved by the two works show that the task of learning through mobile devices can be eased by exploiting HCI design methods.

The results achieved by the proposed systems inspired different enhancements and future works. The museum application is at its first stage of the software engineering process and new features will be introduced to enhance the navigation and the content display tasks. A painting geo-localization system and the 3D techniques used in the image gallery application will be exploited and merged to display 3D contents to the users. Supported by the interesting results achieved by the two described works, the AVIRES team is now leading other projects that involve the mobile learning, especially for museums and exhibits. “The Pantheon” is a novel mobile-based application focused on the famous artist Antonio De’ Sacchis. Information about the artists and their works, together with connected points of interests (e.g. museums and churches), are provided with the goal to improve and ease the visitor’s tasks and to provide relevant and connected information about different works. The proposed application is conceived to be available for both smartphones and tablets, thus introducing different HCI challenges.

17.7. Bibliography

- [BEV 94] BEVAN N., MACLEOD M., “Usability measurement in context”, *Behaviour & Information Technology*, vol. 13, no. 1–2, pp. 132–145, 1994.

- [BOR 97] BORLUND P., INGWERSEN P., “The development of a method for the evaluation of interactive information retrieval systems”, *Journal of Documentation*, vol. 53, no. 3, pp. 225–250, 1997.
- [BRO 06] BROWN M., LOWE D.G., “Automatic panoramic image stitching using invariant features”, *International Journal of Computer Vision*, vol. 74, no. 1, pp. 59–73, 2006.
- [BUR 07] BURIGAT S., CHITTARO L., “Geographical data visualization on mobile devices for user’s navigation and decision support activities”, in BELUSSI A., CATANIA B., CLEMENTINI E., FERRARI E. (eds.), *Spatial Data on The Web*, Springer, Berlin, Heidelberg, pp. 261–284, 2007.
- [BUT 10] BUTTUSSI F., CHITTARO L., “Smarter Phones for Healthier Lifestyles: An Adaptive Fitness Game”. *IEEE Pervasive Computing*, vol. 9, no. 4, pp. 51–57, 2010.
- [CAR 07] CARMO M.B., AFONSO A.P., MATOS P.P., “Visualization of geographic query results for small screen devices”. *ACM workshop on Geographical Information Retrieval*, New York, New York, USA: ACM Press, pp. 63–64.
- [CHE 00] CHEN C., YU Y., “Empirical studies of information visualization: a meta-analysis”, *International Journal of Human-Computer Studies*, vol. 53, no. 5, pp. 851–866, 2000.
- [CHI 11] CHITTARO L., CARCHIETTI E., DE MARCO L., *et al.*, “Personalized emergency medical assistance for disabled people”, *User Modeling and User-Adapted Interaction*, vol. 21, no. 4–5, pp. 407–440, 2011.
- [DAH 02] DAHNE P., KARIGIANNIS J., “Archeoguide: system architecture of a mobile outdoor augmented reality system”, *Proceedings of the International Symposium on Mixed and Augmented Reality*, IEEE Computer Society, pp. 263–264, 2002.
- [DIC] DICTIONARY.COM. (N.D.). “Smartphone definition”. Retrieved from <http://dictionary.reference.com/browse/smartphone>.
- [FAU 03] FAULKNER L., “Beyond the five-user assumption: benefits of increased sample sizes in usability testing”, *Behavior Research Methods, Instruments, & Computers*, vol. 35, no. 3, pp. 379–383, 2003.
- [FIN 12] FINOCCHI V., RADETIC D., “Learning through art history: the multimedia centre and visual art lab about pordenone”, *The Transformative Museum*, Roskilde, Denmark, May 2012.
- [FOR 05] FORESTI G., MICHELONI C., SNIDARO L., *et al.*, “Active video-based surveillance system: the low-level image and video processing techniques needed for implementation”, *IEEE Signal Processing Magazine*, vol. 22, no. 2, pp. 25–37, 2005.

- [FOR 09] FORESTI G.L., MICHELONI C., PICIARELLI C., *et al.*, “Visual sensor technology for advanced surveillance systems: historical view, technological aspects and research activities in Italy”, *Sensors*, vol. 9, no. 4, pp. 2252–2270, 2009.
- [GAL] GALANI A., CHALMERS M., “Far away is close at hand: shared mixed reality museum experiences for local and remote museum companions”, *International Cultural Heritage Informatics Meeting – ICHIM*, Paris, France, pp. 1–15, 2003.
- [GAR 11] GARTNER, Press releases 2011 – mobile phones, <http://www.gartner.com/newsroom/id/1764714>.
- [GJE] GJEDDE L., “The Flex-learn project: designing a mobile learning platform for truck drivers”. Paper presented at *Mobile Monday Conference on Mobile Learning*, Copenhagen, May 2008.
- [HAB 06] HABIB M.K., “Mechatronics engineering the evolution, the needs and the challenges”, *Proceedings of the IECON 2006 – 32nd Annual Conference on IEEE Industrial Electronics*, IEEE, pp. 4510–4515, November 2006.
- [HAB 08] HABIB M.K., “Interdisciplinary Mechatronics engineering and science: problem-solving, creative-thinking and concurrent design synergy”, *International Journal of Mechatronics and Manufacturing Systems*, vol. 1, no. 1, pp. 4–22, 2008.
- [IDC 12] IDC, “IDC increases media tablet forecast, expects new products and price points to drive strong shipment growth through 2016”, http://www.idc.com/getdoc.jsp?containerId=prUS23543712#.URA027_K7ng, 2012.
- [KJE 04] KJELDSKOV J., STAGE J., “New techniques for usability evaluation of mobile systems”, *International Journal of Human-Computer Studies*, vol. 60, no. 5–6, pp. 599–620, 2004.
- [MAR 12] MARTINEL N., MICHELONI C., “Re-identify people in wide area camera network”, *Proceedings of the 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, IEEE, Providence, RI, pp. 31–36, June 2012.
- [MIC 12] MICHELONI C., RANI A., KUMAR S., *et al.*, “A balanced neural tree for pattern classification”, *Neural networks The Official Journal of the International Neural Network Society*, vol. 27, Elsevier, pp. 81–90, March 2012.
- [NIE 93] NIELSEN J., LANDAUER T.K., “A mathematical model of the finding of usability problems”, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems – CHI '93*, ACM Press, New York, pp. 206–213, 1993.
- [NIE 94] NIELSEN J., “Usability inspection methods”, *Proceedings of the Conference Companion on Human Factors in Computing Systems – CHI '94*, ACM Press, New York, pp. 413–414, 1994.
- [NIE 12a] NIELSEN WIRE, “Survey: new U.S. smartphone growth by age and income”, http://blog.nielsen.com/nielsenwire/online_mobile/survey-new-u-s-smartphone-growth-by-age-and-income/, 2012.

- [NIE 12b] NIELSEN WIRE, “Two thirds of new mobile buyers now opting for smartphones”, http://blog.nielsen.com/nielsenwire/online_mobile/two-thirds-of-new-mobile-buyers-now-opting-for-smartphones/, 2012.
- [NIX 05] NIX J., “The Development of Mobile Learning for Smartphones”, *International Conference Mobile Learning, IADIS 2005*, February 22–25, Algarve, Portugal, pp. 118–125, 2005.
- [PIC 09] PICIARELLI C., MICHELONI C., FORESTI G., “PTZ camera network reconfiguration”, *Third ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC)*, pp. 1–7, August 30–September 22, Como, Italy, 2009.
- [PIE 08] PIERROUX P., “Extending meaning from museum visits through the use of wikis and mobile blogging”, *International Conference of the Learning Sciences*, 2008.
- [ROB 08] ROBBINS, D.C., LEE, B., FERNANDEZ, R., “TapGlance: designing a unified smartphone interface”, *Proceedings of the 7th ACM Conference on Designing interactive systems – DIS ’08*, New York, USA, pp. 386–394, 2008.
- [ROS 03] ROSCHELLE J., “Wireless mobile devices”, vol. 19, pp. 260–272, 2003.
- [SAM 06] SAMPSON D., “Exploiting mobile and wireless technologies in vocational training”, *2006 4th IEEE International Workshop on Wireless, Mobile and Ubiquitous Technology in Education (WMTE’06)*, IEEE, pp. 63–65, November 2006.
- [SHA 00] SHARPLES M., “The design of personal mobile technologies for lifelong learning”, *Computers & Education*, vol. 34, no. 3–4, pp. 177–193, 2000.
- [SCH 06] SCHRADER K., NGUYEN-DOBINSKY T.-N., KAYSER K., *et al.*, “Mobile education in autopsy conferences of pathology: presentation of complex cases”, *Diagnostic Pathology*, vol. 1(42), pp. 1–8, 2006.
- [SMØ 03] SMØRDAL O., GREGORY J., “Personal digital assistants in medical education and practice”, *Journal of Computer Assisted Learning*, vol. 19, pp. 320–329, 2003.
- [STO 05] STOICA A., FIOTAKIS G., CABRERA J.S., *et al.*, “Usability evaluation of handheld devices: A case study for a museum application”, *Panhellenic Conference on Informatics*, Volos, Greece, November 11–15, 2005.
- [TAK 08] TAKACS G., CHANDRASEKHAR V., GELFAND N., *et al.*, “Outdoors augmented reality on mobile phone using loxel-based visual feature organization”, *Proceeding of the 1st ACM International Conference on Multimedia Information Retrieval (MIR ’08)*, New York, USA: ACM Press, pp. 427–434, 2008.
- [TRA 05] TRAXLER J., “Defining mobile learning”, *IADIS International Conference Mobile Learning*, Qawra, Malta, pp. 261–266, June 28–30, 2005.
- [VAV 09] VAVOULA G., SHARPLES M., RUDMAN P., *et al.*, “Myartspace: design and evaluation of support for learning with multimedia phones between classrooms and museums”, *Computers & Education*, vol. 53, no. 2, pp. 286–299, 2009.

- [VLA 01] VLAHAKIS V., IOANNIDIS N., KARIGIANNIS J., *et al.*, “Archeoguide”, *Proceedings of the 2001 Conference on Virtual Reality, Archeology, and Cultural Heritage – VAST 01*, New York, USA: ACM Press, p. 131, 2001.
- [VLA 02] VLAHAKIS V., IOANNIDIS M., KARIGIANNIS J., *et al.*, “Archeoguide: an augmented reality guide for archaeological sites”, *IEEE Computer Graphics and Applications*, vol. 22, no. 5, pp. 52–60, 2002.
- [W3C 05] W3C, “Mobile web initiative device description working group charter (DDWG)”, 2005.
- [WEI 99] WEISER M., “The computer for the 21st century”, *ACM SIGMOBILE Mobile Computing and Communications Review – Special Issue dedicated to Mark Weiser*, vol. 3, no. 3, pp. 3–11, 1999.
- [ZHE 05] ZHENG P., NI L., *Smart Phone and Next Generation Mobile Computing*, 1st ed., Morgan Kaufmann, 2005.

Chapter 18

Application of Stereo Vision and ARM Processor for Motion Control

The main idea of this chapter is to present the application of vision-based approach via stereo vision to determine position and orientation of mechanical manipulators. Stereo vision is a technique based on inferring the depth of an object with the aim of two cameras. In this case, a stereo vision setup that is made up of two parallel complementary metal oxide semiconductor (CMOS) cameras is provided and Universal Serial Bus (USB) port is used to transfer image data to a computer. Experimental tests are exerted for 6R mechanical manipulator in point-to-point motion. The image processing algorithm, matching algorithm and triangulation calculations are optimized to increase the speed of computations that are required for the matching procedure. Because of the rapid image processing algorithm, the proposed method can be used to detect the end-effector in dynamic trajectories. This stereo vision setup can be applied easily as a measurement system in closed-loop control systems or other applications.

18.1. Introduction

Determining the real end-effector position for mechanical manipulators is an important key point in industrial applications. This problem can be

considered as the process of determining the position and orientation of the end-effector of robotic manipulators with respect to a global frame. Various methods and solutions have been proposed and applied to this problem in robotic areas recently. The possible easiest solution to this problem by is doing relative position/orientation measurements using sensors such as odometer or an inertia measurement unit (IMU). A relative position measurement known as dead reckoning is the process of tracking the current position of the end-effector based on the integration of the path that has been previously traveled by the end-effector [HEE 07]. The most common sensors are IMU which is made of accelerometers and gyroscopic rate sensors. IMUs have the advantage of a high frame rate and relatively small latency, but only provide relative motion and they are subject to significant drift (or significant cost). IMU sensors are often complemented by global positioning system (GPS) sensors [JON 10]. In dead reckoning, odometry is probably the most well-known method that is used in mobile robots, as it provides easy and cheap real-time position information by the integration of incremental motion information over time. Unfortunately, this integration causes an accumulation of errors during the movement of the robot [GUT 08].

Another approach for an end-effector position measurement is by using active beacons. Measurement using active beacons has traditionally been used in the GPS for the localization of ships and airplanes [HEE 07]. There are two types of active beacon systems: trilateration and triangulation [KUC 06, TSU 09, SAN 09]. Trilateration is the determination of the end-effector position based on distance measurements to known beacon sources. In trilateration systems, there are usually three or more transmitters mounted at known locations in the workspace and one receiver fixed onto the end-effector. An example of the active beacons localization system that makes use of trilateration is given in [MOO 04].

Triangulation is another way of determination of the robot end-effector position that is based on the measurement of the angle from the beacon to the end-effector or robot heading. The distance of at least one of the beacons and its location must be known. Similar to the trilateration method, three or more beacon readings must be obtained to do triangulation. Note that it is also possible to do triangulation with two beacons if the angles from these beacons to the end-effector or robot heading, the distances from these beacons to the robot and the locations of these beacons are known [EVO]. An example of the triangulation with two beacons is the North Star kit [EVO]. Unlike dead reckoning, the errors in active beacons systems will not grow unbounded; however, accuracy is highly dependent on the size of its random

errors and precise position of the beacons in the environment [HEE 07].

For compensating the disadvantages of relative measurement methods, sensors that provide measurements of absolute position are extremely important. These kind of methods are named absolute position measurements (reference-based systems). Magnetic compasses and vision systems [RAM 08, CHA 10, SPA 07] are two approaches of this category. Chang *et al.* [CHA 10] presented a vision-based navigation and localization system using two biologically inspired scene understanding models that are studied from human visual capabilities' point of view: (1) a Gist model that captures the holistic characteristics and layout of an image and (2) a saliency model that emulates the visual attention of primates to identify conspicuous regions in the image.

The main idea of this chapter is to use the stereo vision approach and propose a simple triangulation approach for determining position and orientation using a simple and inexpensive setup for mechanical manipulators. On the basis of stereo vision, a dynamical test is also implemented for a 6R manipulator to show the accuracy and speed of computations of the proposed approach. Stereo vision represents a way for reconstructing three-dimensional (3D) information from the surrounding environment and it is important for a large number of vision applications [BLE 08]. Unfortunately, the key step in stereo vision, that is the stereo matching problem, cannot be regarded as solved. Other factors that complicate the matching process are image noise, untextured regions and the occlusion problem [BLE 08]. Two methods are used for the matching process: local and global methods [HUM 10, YAN 06].

Stan and Carlo [STA 99] proposed an algorithm to detect depth discontinuities from a stereo pair of images. The algorithm matched individual pixels in corresponding scan line pairs while allowing occluded pixels to remain unmatched, and then propagated the information between scan lines by means of a fast post-processor. The algorithm handled large untextured regions, used the measurement of pixel dissimilarity that was insensitive to image sampling and pruned bad search nodes to increase the speed of dynamic programming. The computation was relatively fast, taking about 1.5 μ s per pixel per disparity on a workstation. Approximate disparity maps and precise depth discontinuities (along both horizontal and vertical boundaries) were shown for five stereo images.

18.2. Stereo vision

The three-dimensional measurement approach based on the human vision system is named stereo vision, but in this vision approach, human eyes are replaced with a pair of slightly displaced parallel cameras. Two-dimensional position information of an object is determined using one camera and a third coordinate can be computed by comparing two images by means of a technique based on stereo vision approach. This technique helps to infer the depth of an object from two cameras.

Using stereo vision, with two images, we can infer depth by means of the triangulation method if it is available to find corresponding (homologous) points in the two images [VIS]. According to Figure 18.1, the amount to which a single pixel is displaced in the two images is called disparity, so a pixel's disparity is inversely proportional to its depth in the scene. Before computing disparity, a rectification process must be applied to remove lens distortions and turn the stereo pair in to standard form.

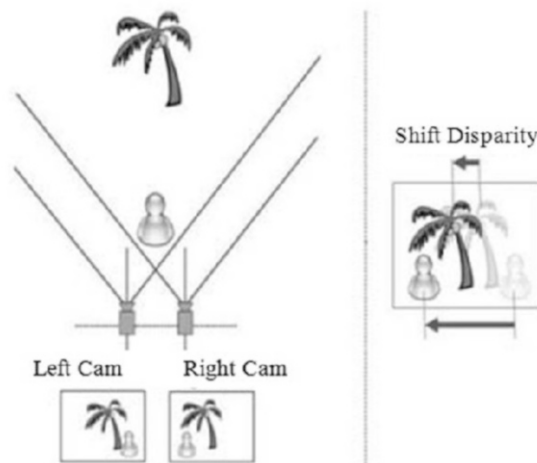


Figure 18.1. Relation between depth and disparity [DAS 10]

The challenging part is to compute the disparity of each pixel and this task is known as the stereo matching problem. Two methods are used for the matching process: local and global methods and local and global/semi-global methods. Local algorithms use the simple winner-takes-all (WTA) disparity selection strategy but reduce the ambiguity (increasing the signal-to-noise

(SNR) ratio) by aggregating matching costs over a support window (aka kernel or correlation window). Global (and semi-global) algorithms search for disparity assignments that minimize an energy function over the whole stereo pair using a pixel-based matching cost (sometimes the matching cost is aggregated over a support) [VIS].

The Middlebury stereo evaluation site provides a framework and a data set for benchmarking novel algorithms [SCH]. An overview of the stereo vision approach is shown in Figure 18.2.

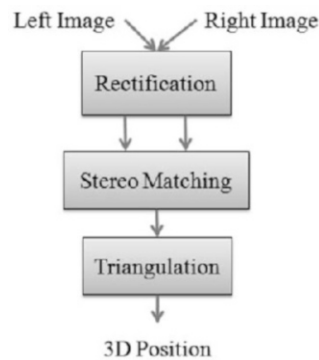


Figure 18.2. Overview of stereo vision approach

18.3. Triangulation

Given the disparity of any pixel, triangulation computes the position of the correspondence in the 3D space. Consider Figure 18.3 that shows two images, an object, the image frame (local) and the reference frame. Parameters that are shown in this figure are:

- OXYZ: reference frame;
- $O_1x_1y_1z_1$: image frame of first camera;
- $O_2x_2y_2z_2$: image frame of second camera;
- F_1, F_2 : focal lengths;
- β_1, β_2 : rotation angles of optical axis;
- u_1, v_1 : pixel coordinate in first image;
- u_2, v_2 : pixel coordinate in second image.

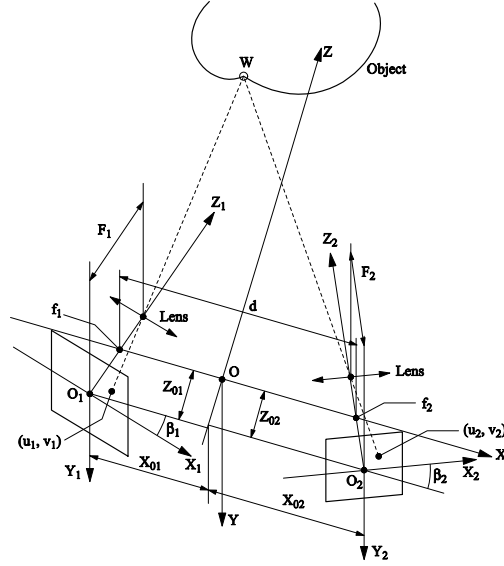


Figure 18.3. *Coordinates in stereo vision [DAS 10]*

Transformation between the object position in the reference frame and each local frame can be stated as:

$$M^T = RTP^T \quad [18.1]$$

where P and M are the local and the global positions of the object and have the following forms:

$$P = [x, y, z, 1] \quad [18.2]$$

$$M = [X, Y, Z, 1] \quad [18.3]$$

and R and T matrices in equation [18.1] are rotation and translation matrixes, respectively, and have the following forms:

$$R = \begin{bmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [18.4]$$

$$T = \begin{bmatrix} 1 & 0 & 0 & X_0 \\ 0 & 1 & 0 & Y_0 \\ 0 & 0 & 1 & Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [18.5]$$

In these equations, β is the rotation of an optical axis of the considered camera and $[X_0, Y_0, Z_0]$ are the position of the center of the local frame in the reference frame. Consider F as focal length and as a result, perspective relations for any image can be written as:

$$\frac{x}{u} = -\frac{z-F}{F}; \quad \frac{y}{v} = -\frac{z-F}{F} \quad [18.6]$$

Substituting equation [18.6] into equation [18.1], for two images, the 3D position of an object in the reference frame can be computed as:

$$\begin{aligned} X &= \frac{u_1 F_1 - m_1 X_{01} - n_1 (Z - Z_{01})}{m_1} \\ Y &= v_1 - Y_{01} - \frac{v_1}{F_1} [(X - X_{01}) \sin \beta_1 + (Z - Z_{01}) \cos \beta_1] \\ Z &= \frac{m_1 m_2 (X - X_{01}) + m_1 F_2 u_2 - m_2 F_1 u_1 + m_2 n_1 Z_{01} - m_1 n_2 Z_{02}}{m_1 n_2 - m_2 n_1} \end{aligned} \quad [18.7]$$

in which

$$\begin{aligned} m_1 &= F_1 \cos \beta_1 + u_1 \sin \beta_1 \\ m_2 &= F_2 \cos \beta_2 - u_2 \sin \beta_2 \\ n_1 &= u_1 \cos \beta_1 - F_1 \sin \beta_1 \\ n_2 &= u_2 \cos \beta_2 + F_2 \sin \beta_2 \end{aligned} \quad [18.8]$$

where $[u_i, v_i]$ are the coordinates of the projection of the object into i th image. If two cameras could be parallel with $\beta = 0$, then the following simple relations can be considered:

$$\begin{aligned} Y_{01} &= Y_{02} = 0 \\ \beta_1 &= \beta_2 = 0 \\ F_1 &= F_2 = F \\ X_{01} &= X_{02} = \frac{B}{2} \end{aligned} \quad [18.9]$$

Finally, equation [18.7] can be rewritten as:

$$\begin{aligned} X &= \frac{F-Z}{F}u_1 - \frac{B}{2} \\ Y &= \frac{F-Z}{F}v_1 \\ Z &= F - \frac{BF}{u_2 - u_2} \end{aligned} \quad [18.10]$$

In equation [18.10], $[u_1, u_2]$ are the differences between the X coordinate of two corresponding pixels that were named disparity. The disparity of an object that is in infinity is equal to zero. According to the above equations, parameters that affect the accuracy of measurements are:

- 1) accuracy of stereo correspondence;
- 2) distance between cameras and end-effector;
- 3) initial calibration.

18.4. End-effector orientation

According to the triangulation method and using the stereo vision setup, the position of the end-effector and coordinates can be calculated, which are shown in Figure 18.4 as P_1 , P_2 and P_3 . The direction vectors $\vec{x}_b$, $\vec{y}_b$ and $\vec{z}_b$ can be represented as:

$$\vec{x}_b = \frac{P_2 - P_1}{\|P_2 - P_1\|} \quad [18.11]$$

$$\vec{z}_b = \frac{(P_2 + P_1)/2 - P_3}{\|(P_2 + P_1)/2 - P_3\|} \quad [18.12]$$

$$\vec{y}_b = \vec{z}_b \times \vec{x}_b \quad [18.13]$$

Considering the rotation matrices $R_{Z,\phi}$, $R_{Y,\theta}$ and $R_{X,\psi}$ for roll, pitch and yaw angles, the rotation matrix between reference and body frames can be mentioned as:

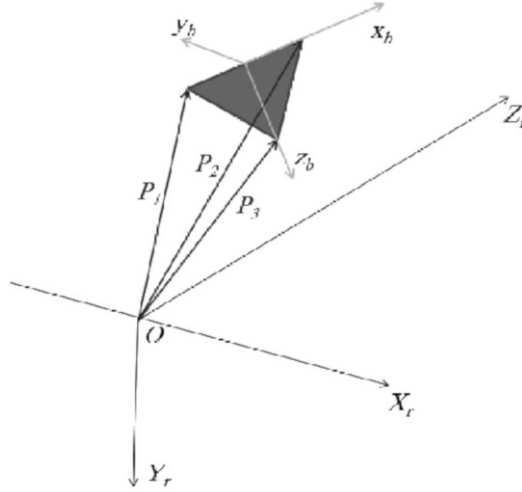


Figure 18.4. Three points on end-effector

$$\begin{bmatrix} \bar{x}_b & \bar{y}_b & \bar{z}_b \end{bmatrix} = \begin{bmatrix} X_x & Y_x & Z_x \\ X_y & Y_y & Z_y \\ X_z & Y_z & Z_z \end{bmatrix} \quad [18.14]$$

$$\begin{bmatrix} X_x & Y_x & Z_x \\ X_y & Y_y & Z_y \\ X_z & Y_z & Z_z \end{bmatrix} = \begin{bmatrix} C_\phi C_\theta & C_\phi S_\theta S_\psi - S_\phi C_\psi & C_\phi S_\theta C_\psi + S_\phi S_\psi \\ S_\phi C_\theta & S_\phi S_\theta S_\psi - C_\phi C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi \\ -S_\theta & C_\theta S_\psi & C_\theta C_\psi \end{bmatrix} \quad [18.15]$$

where $C_i = \cos(i)$ and $S_i = \sin(i)$. Therefore, the roll, pitch and yaw angles of the robot end-effector are determined as follows:

$$\begin{aligned} \phi &= \arctan\left(\frac{X_y}{X_x}\right) \\ \theta &= \arctan\left(\frac{-X_z}{\sqrt{1 - X_z^2}}\right) \\ \psi &= \arctan\left(\frac{Y_z}{Z_z}\right) \end{aligned} \quad [18.16]$$

Figure 18.3 may seem complicated, but with the above derivations the coordinates can be computed easily. One of the advantages of using measurements based on vision in different systems is the simple equations that determine the position of an object.

18.5. Experimental setup and results

To show the application of the vision system, an experimental test is conducted on a 6R manipulator. The robot arm is shown in Figure 18.5. This robot arm is an articulated six degree of freedom (DoF) arm that has the base of a programmable universal machine for assembly (PUMA) structure and a three DoF end-effector. The six DoFs of the 6R provide the motion in a 3D space with the path and the orientation of the gripper.



Figure 18.5. *6R robot arm*

The schematic of the manipulator is shown in Figure 18.6. The coordinate frames on each joint are set based on the Denavit-Hartenberg algorithm to prepare the transformation matrices for calculation of direct and inverse kinematics. A distance is considered between fifth and sixth links for better presentation, which is in fact not the reality. The Denavit-Hartenberg parameters of the arm are presented in Table 18.1.

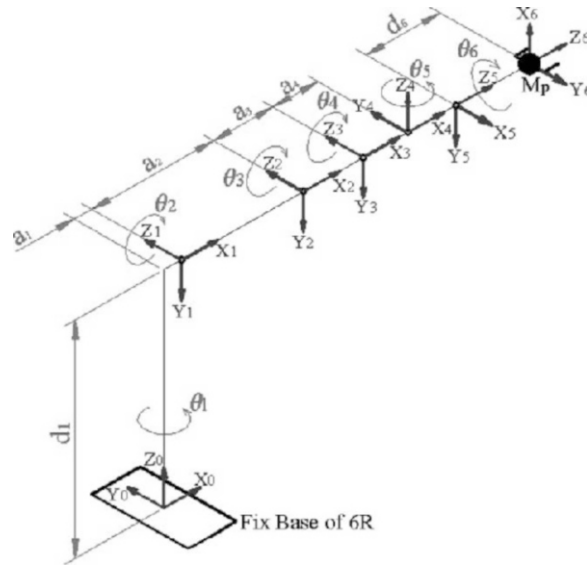


Figure 18.6. Schematic of 6R robot arm

Joint	a_i (mm)	d_i (mm)	α_i^0	Motion	Home position
1	36.5	438	-90	Link 1	0
2	251.5	0	0	Link 2	0
3	125	0	0	Link 3	0
4	92	0	90	Yaw	0
5	0	0	-90	Pitch	-90
6	0	152.8	0	Roll	0

Table 18.1. Denavit-Hartenberg of 6R robot arm

Transformation matrix, T , is used for forward kinematic computations based on multiplying six transformations coordinates to each other.

$$T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [18.17]$$

in which the elements of T are:

$$\begin{aligned}
 n_x &= -c_6 s_1 s_5 + c_1 (c_{234} c_5 c_6 - s_{234} s_6) \\
 n_y &= c_{234} c_5 c_6 s_1 + c_1 c_6 s_5 - s_1 s_{234} s_6 \\
 n_z &= -c_5 c_6 s_{234} - c_{234} s_6 \\
 o_x &= s_1 s_5 s_6 - c_1 (c_6 s_{234} + c_{234} c_5 s_6) \\
 o_y &= -c_6 s_1 s_{234} - (c_{234} c_5 s_1 + c_1 s_5) s_6 \\
 o_z &= -c_{234} c_6 + c_5 s_{234} s_6 \\
 a_x &= -c_5 s_1 - c_1 c_{234} s_5 \\
 a_y &= c_1 c_5 - c_{234} s_1 s_5 \\
 a_z &= s_{234} s_5 \\
 p_x &= -d_6 c_5 s_1 + c_1 (a_1 + a_2 c_2 + a_3 c_{23} + c_{234} (a_4 - d_6 s_5)) \\
 p_y &= d_6 c_1 c_5 + s_1 (a_1 + a_2 c_2 + a_3 c_{23} + c_{234} (a_4 - d_6 s_5)) \\
 p_z &= d_1 - a_2 s_2 - a_3 s_{23} + s_{234} (-a_4 + d_6 s_5)
 \end{aligned} \tag{18.18}$$

In these equations, a_i and d_i are shown in Figure 18.6, also s_i , c_i , s_{ij} and c_{ij} denote $\sin(\theta_i)$, $\cos(\theta_i)$, $\sin(\theta_i + \theta_j)$ and $\cos(\theta_i + \theta_j)$, respectively. P_x , P_y and P_z are the end-effector position of 6R. The relation between velocity of end-effector and angular velocity of joints is expressed by:

$$V = J\dot{q} \tag{18.19}$$

J is the Jacobian matrix of the 6R robot and can be obtained as:

$$J = \begin{bmatrix} \dot{j}_{11} & \dot{j}_{12} & \dot{j}_{13} & \dot{j}_{14} & \dot{j}_{15} & 0 \\ \dot{j}_{21} & \dot{j}_{22} & \dot{j}_{23} & \dot{j}_{24} & \dot{j}_{25} & 0 \\ 0 & \dot{j}_{32} & \dot{j}_{33} & \dot{j}_{34} & d_6 c_5 s_{234} & 0 \\ 0 & -s_1 & -s_1 & -s_1 & c_1 s_{234} & \dot{j}_{46} \\ 0 & c_1 & c_1 & c_1 & s_1 s_{234} & \dot{j}_{56} \\ 1 & 0 & 0 & 0 & c_{234} & s_{234} s_5 \end{bmatrix} \tag{18.20}$$

where

$$\begin{aligned}
 j_{11} &= -d_6 c_1 c_5 - s_1 (a_1 + a_2 c_2 + a_3 c_{23} + c_{234} (a_4 - d_6 s_5)) \\
 j_{12} &= -c_1 (a_2 s_2 + a_3 s_{23} + s_{234} (a_4 - d_6 s_5)) \\
 j_{13} &= -c_1 (a_3 s_{23} + s_{234} (a_4 - d_6 s_5)) \\
 j_{14} &= c_1 s_{234} (-a_4 + d_6 s_5) \\
 j_{15} &= -d_6 c_1 c_{234} c_5 + d_6 s_1 s_5 \\
 j_{21} &= -d_6 s_1 c_5 + c_1 (a_1 + a_2 c_2 + a_3 c_{23} + c_{234} (a_4 - d_6 s_5)) \\
 j_{22} &= -s_1 (a_2 s_2 + a_3 s_{23} + s_{234} (a_4 - d_6 s_5)) \\
 j_{23} &= -s_1 (a_3 s_{23} + s_{234} (a_4 - d_6 s_5)) \\
 j_{24} &= s_1 s_{234} (-a_4 + d_6 s_5) \\
 j_{25} &= -d_6 (c_{234} c_5 s_1 + c_1 s_5) \\
 j_{32} &= -a_2 c_2 - a_3 c_{23} + c_{234} (-a_4 + d_6 s_5) \\
 j_{33} &= -a_3 c_{23} + c_{234} (-a_4 + d_6 s_5) \\
 j_{34} &= c_{234} (-a_4 + d_6 s_5) \\
 j_{46} &= -c_5 s_1 - c_1 c_{234} s_5 \\
 j_{56} &= c_1 c_5 - c_{234} s_1 s_5
 \end{aligned} \tag{18.21}$$

The trajectory tracking cases need the Jacobian matrix to compute the desired angles from the end-effector path. The controller of the robot is proportional-integral derivative (PID), which is implemented on a board with an ARM microprocessor (Figure 18.7(a)). The vision system that consists of two high definition (HD) cameras, parallel to each other, is presented in Figure 18.7(b).

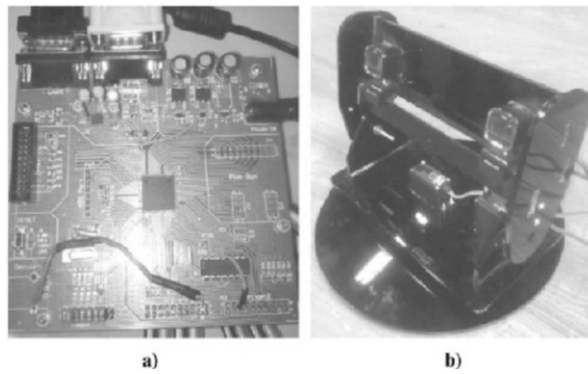


Figure 18.7. a) Digital board with ARM processor and b) vision system

By applying the PID controller, the result of the point-to-point motion is obtained. In Figure 18.8, the variations of angles of links are shown that are the results of a potentiometer measurement. Figure 18.9 shows the 3D path of the end-effector.

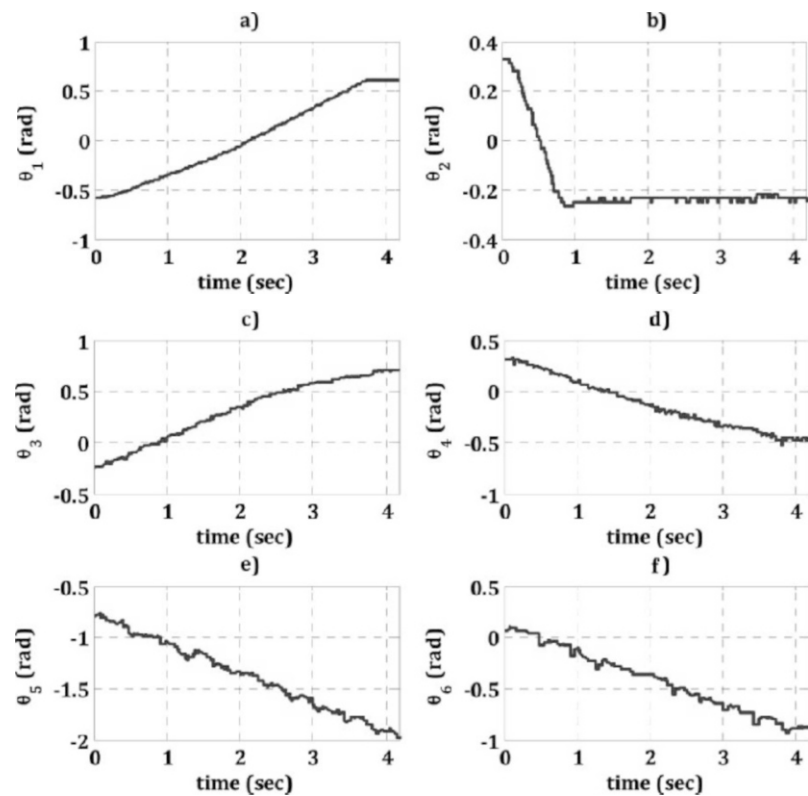


Figure 18.8. The variations of angles of links 1–6

In some cases, as it is for this case, the measurements based on the variation of angles of links and then the calculation of the end-effector of the arm is not the best way for an analysis of motion. The inaccuracy is due to improper modeling, backlash, etc. External measurements such as vision can provide an independent result that helps even to check the previous measurement system.

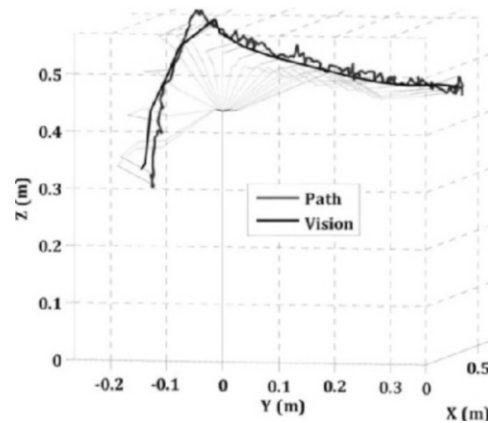


Figure 18.9. End-effector trajectory via potentiometer and vision measurements

18.6. Summary

The focus of this chapter was the application of a stereo vision approach on robotic systems to measure the 3D position of the end-effector during motion. Stereo vision is a method to determine the 3D position information of an object. A simple and inexpensive setup, containing two cameras, is made to implement stereo vision on any mechanical manipulator. Also a basic formulation of the triangulation process is mentioned to compute the 3D position and orientation of the end-effector based on two resulted images. Experimental results are provided for a 6R mechanical manipulator to show the accuracy of the proposed method for end-effector positioning. According to the experimental output, the following results can be mentioned:

- 1) The performance of the measurement system is related to the quality of the cameras and it can affect the accuracy and speed of the stereo vision system.
- 2) The other main factor is the distance between setup and the end-effector. If this distance is decreased, then the accuracy of the positioning system will be increased.
- 3) The sampling frequency of the measurement is a function of the quality and speed of the cameras.
- 4) A 3D position that is determined via stereo vision can be used in a closed-loop control system. But high sampling frequency is needed.

18.7. Bibliography

- [BLE 08] BLEYER M., GELAUTZ M., “Simple but effective tree structures for dynamic programming-based stereo matching”, *Computer Vision Theory and Applications, VISAPP*, Funchal, Madeira, Portugal, pp. 415–422, 2008.
- [CHA 10] CHANG C.K., SIAGIAN C., ITTI L., “Mobile robot vision navigation and localization using gist and saliency”, *IEEE International Conference on Intelligent Robots and Systems*, Taiwan, pp. 4147–4154, 2010.
- [DAS 10] DASTJERDI A.H., Optimizing tracking algorithm for contact probe of CMM by non contact probe based on image processing, MSc Thesis, Amir Kabir University, Iran, 2010.
- [EVO] EVOLUTION ROBOTICS INC., *North Star Projector Kit User Guide*, 1.0 ed., CA. <https://www.irobot.com/>
- [GUT 08] GUTIERREZ A., CAMPO A., L. SANTOS F.C., *et al.*, *Social Odometry in Populations of Autonomous Robots*, Lecture Notes in Computer Science, vol. 5217, pp. 371–378, 2008.
- [HEE 07] HEE L.J., Sensor based localization of a mobile, PhD Thesis, National University of Singapore, 2007.
- [HUM 10] HUMENBERGER M., ENGELKE T., KUBINGER W., “A census-based stereo vision algorithm using modified semi-global matching and plane fitting to improve matching quality”, *IEEE Computer Society Conference CVPRW*, San Francisco, CA, pp. 77–84, 2010.
- [JON 10] JONES E., SOATTO S., “Visual-inertial navigation, mapping and localization: a scalable real-time causal approach”, *International Journal of Robotics Research*, vol. 30, pp. 407–430, 2010.
- [KUC 06] KUCSERA P., “Sensors for mobile robot systems”, *Academic and Applied Research in Military Science*, vol. 5, no. 4, pp. 645–658, 2006.
- [MOO 04] MOORE D., LEONARD J., RUS D., *et al.*, “Robust distributed network localization with noisy range measurements”, *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, ACM, Baltimore, MD, USA, 2004.
- [RAM 08] RAMISA A., TAPUS A., MANTARAS R.L., *et al.*, “Mobile robot localization using panoramic vision and combinations of feature region detectors”, *IEEE International Conference on Robotics and Automation*, Pasadena, CA, pp. 538–543, 2008.
- [SAN 09] SANTOLARIA J., GUILLOMÍA D., CAJAL C., *et al.*, “Modeling and calibration technique of laser triangulation sensors for integration in robot arms and articulated arm coordinate measuring machines”, *Sensors*, vol. 9, pp. 7374–7396, 2009.

- [SCH] Middlebury Stereo Evaluation, available at <http://vision.middlebury.edu/stereo/eval/>
- [SPA 07] SPACEK L., BURBRIDGE C., “Instantaneous robot self-localization and motion estimation with omnidirectional vision”, *Robotics and Autonomous Systems*, vol. 55, pp. 667–674, 2007.
- [STA 99] STAN B., CARLO T., “Depth discontinuities by pixel-to-pixel stereo”, *International Journal of Computer Vision*, vol. 35, no. 3, pp. 269–293, 1999.
- [TSU 09] TSUKIYAMA T., “Mobile robot localization from landmark bearings”, *World Congress Fundamental and Applied Metrology*, 6–11 September, Lisbon, Portugal, 2009.
- [VIS] University of Bologna, <http://www.vision.deis.unibo.it/smatt/>
- [YAN 06] YANG Q., “Real-time global stereo matching using hierarchical belief propagation”, *British Machine Vision Conference*, Edinburgh, Scotland, pp. 989–998, 2006.

Chapter 19

Mechatronics as Science and Engineering – or Both

19.1. Introduction

The Internet upheaval is established in the 21st Century. This has affected a number of new areas of science and engineering. Basic engineering ancestry has begun to grow into a second-level approach. Most of the imaginary “things” would now become true with effective and heavy mathematical tools. This is because of new technological advancements in human society and their innovative creations and thinking for the next century. Today’s scientists and engineers are sure to join our club to celebrate the outcome.

We begin here with the theories of mechatronics as science and, at the same time, as engineering studies and topics. We flavor some of our experiments with the real-world of businesses. We explain how we did it and why. In this respect we would invite you to go through our initiated cases to the main body of this chapter. The following chapters introduce experiments and case studies which are implemented in the Finnish industrial environment. We believe that it is practical and a learning experience for the readers of this handbook today and for the future.

Let us first become acquainted and established with mechatronic systems in more detail. The encyclopedia of Wikipedia¹ has defined that a mechatronics engineer, is able to blend the principles of mechanics, electronics and computing; to create something simpler, while adding value and a reliable system. It is also centered on mechanics, electronics, computing, control engineering, molecular engineering (from nanochemistry and biology) and optical engineering. Through a group effort, the mechatronic modules perform the production goals and inherit flexible and agile manufacturing properties in the production scheme. Figure 19.1² shows an enlarged version of the interconnectivity to grasp the veracity of mechatronics in general.

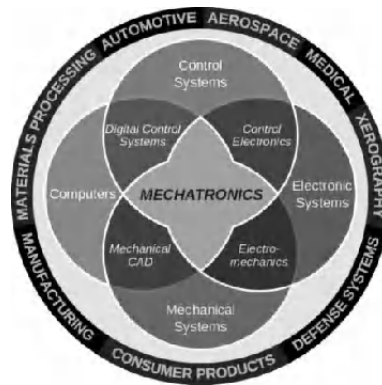


Figure 19.1. *Veracity of mechatronics*

In other words, mechatronics engineering and science is mastered in almost all the sectors that we know so far [HAB 07]. Therefore, it is now time for us to enter into, with a bit more detail but with limitations, the field of mechatronic products³, its development schemes and implementation methodology scenarios. The engineering of cybernetics welds with the question of control engineering of mechatronic systems. Modern production machines consist of mechatronic modules that are integrated according to the

¹ www.wikipedia.org.

² Adapted from the Aerial Euler diagram from RPI's Website, the figure describes the various fields that combine the mechatronics engineering.

³ The portmanteau "mechatronics" was coined by Tetsuro Mori, a senior engineer of the Japanese company Yaskawa in 1969. An industrial robot or forest harvesting machine is a typical example of a mechatronics system; it includes aspects of electronics, mechanics and computing to do its day-to-day jobs.

control engineering architecture. The most common architectures involve hierarchy, polyarchy, heterarchy and hybrid. The methods for achieving a technical consequence are described by control algorithms, which might or might not use formal methods in their design. Hybrid systems, important to mechatronics, include production systems, synergy drives, planetary exploration rovers, automotive subsystems, such as anti-lock braking systems and spin assist, and everyday equipment, such as autofocus cameras, video, hard disks and CD players.

When we enthusiastically focus on mechatronic products and their service aspects, it is necessary and practicable to know how those are in the first place planned, designed and anchored into the manufacturing environment that delivers the end product to the final customers. On the one hand, the customer could be a middle or intermediate one, and at the same time, it could be a participator in other large systems, such as a paper machine parts or systems, or turbo engines, or, say, passenger ships. Product design masked with precise or specific technical and ergonomic disciplines that are systematically followed by an A to Z⁴ are set at the “idea-phase” on a blackboard for physical product. The product creation generates from “zero-level; that is the idea”, and goes through an immense number of technical and commercial phases [HAB 07, HAB 06], before it could be grounded as a kind of physical product at the process end or this end leads to a second or third level in large systems or even simple and stand-alone systems. In the process, “*the idea*” turns from a “skeleton” to “flesh”. At this phase, there are a number of unanswered or even unquestioned elements that are directly attached to or concerned with the product and its manufacturing environments. This may sometimes lead to, in some sense, the engineering one, constructed on “if-and-then” notions. Occasionally, but not always, the situation required a superior technical touch in setups, and its configuration routes perhaps do not exist or are clearly specified, while the future possibility in modification tracks is missing or demands the usability trajectory failure, maintainability, functionality and operability. Nevertheless, when we take a shortcut directly to design, the design procedures are very pragmatic, which involves multi-scientific and engineering knowledge, a lot of drafting, drifting, modeling, versioning, prototyping including interoperability scheme checking, tests on functionality, performance and so on [HAB 08]. Therefore, it is absolutely necessary to know the state of the art so as to know how the design and planning process are met in systematically going forward until the manufacturing of a typical mechatronic product is in hand.

⁴ An A to Z is namely a set of many actions that are organized in a series to be implemented at sequences.

19.2. Theories and methods of design, planning and manufacturing

We will now figure out the theories and methods including the design patterns, solutions and structures of mechatronic products, a paradigm in its manufacturing point of view and the product release strategies. Figure 19.2 shows the business operations, where manufacturing and its connected activities are clustered into each other.

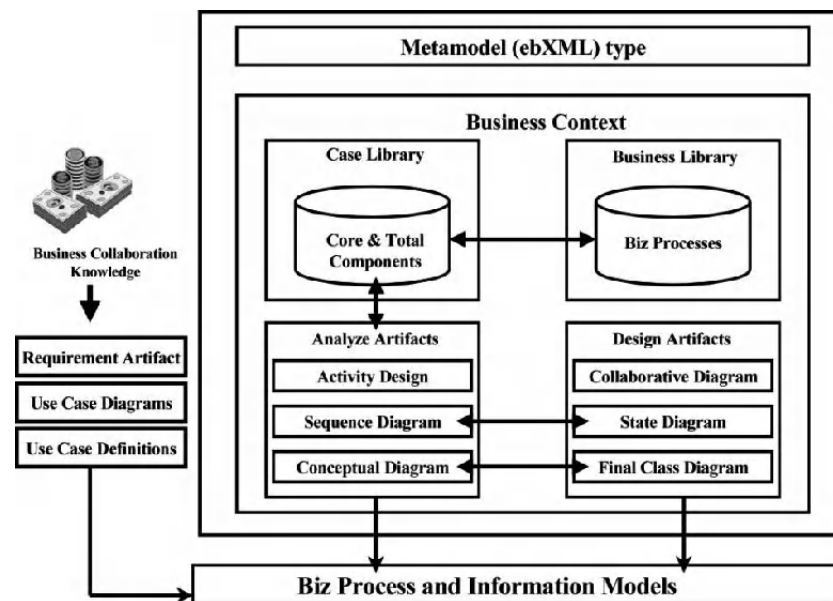


Figure 19.2. Business operations and modeling structure

Figure 19.2 is a typical modeling diagram that shows the entire business process and how its information flows can be modeled on an ebXML⁵ platform. The metamodeling with electronic business concept is not a new one and is implemented in many industrial environments. This system approach would facilitate, perhaps, “the missing-links” between product system architecture and knowledge depositories. In addition, the modeling concept would assist the peers, their spare part-vision targets, or cover the design trajectory envelops and depositories of the data environment.

⁵ ebXML = electronic business extendable mark-up language.

To be successful in business, a company must increase the frequency and speed at which it comes out with new marketable and producible models and constructions. The driving forces behind product development are the demands of customers that are based on their present requirements, and may be the new products supplied by other competitors. There are also other factors that are built in such as involving new investments in manufacturing technology by the main supplier company and also a supplier's capability in improving their overall productivity.

Product design has such a huge effect on the operation and economy of the entire company that more versatile tools must be used for measuring its efficiency. At least the following key factors should be taken into consideration:

- accuracy and relevancy (external and internal);
- meeting the needs of the market;
- keeping the production costs down;
- speed;
- efficiency in the use of research and development (R&D) resources.

It is important to know the interfaces of these factors while modeling the business processes of the company (see Figure 19.3).



Figure 19.3. *Business processes and vision*

When we begin to speed up the product development process, we must bear in mind that we cannot afford to lower the quality of the work and the

reliability of the end product. On the contrary, they should be improved even when we work at a higher speed. Because the majority of the costs related to an engineering product are already determined at the design and planning stage, it is absolutely essential that the actual production costs are kept on a competitive level even when we are developing new products faster. This is only possible if the product designers and the people responsible for production system development work in close cooperation [AND 83]. Generally, the marketing experts, with a deep knowledge of the market requirements, must participate in the development work right from the beginning. These requirements already make intensive teamwork necessary in the future product design work as well. Because the logistics experts and subcontractors must be included at a relatively early stage, the project manager is faced with a really hard task, when the aim is higher development speeds.

The interaction between modules to be connected with each other has an essential impact on the development speed. The module interactions are basically: on, functional, physical, chemical, geometrical and related to the information content. The greatest loss of product development time is caused by the physical, geometrical and informational aspects. At its various generation and creation phases, the product is treated in different ways, advancing from product orientation to the process orientation. At the different phases, the organization will also be different.

Furthermore, we proceed ahead in handling the product development strategies, such as speed and the ergonomic aspects, where we might encounter, sometimes purposely or otherwise, several imaginary or visionary steps for the sake of arguments at executing the product strategy. Normal procedure predicts at this phase of the process or sees more than one complication, or it tends to be complicated by passing a step further into an alternative constructive version that enhances further complexity. Let us see how we could theoretically avoid or survive within such situations.

19.3. Complexity versus complicatedness

Tang and Salminen [TAN 01] illustrated how the complexity versus complicatedness can be handled. They assume in their work that global, dynamic and a competitive business environment has increased the complexity in product, service, operational processes and on the human side. Much engineering effort goes into reducing systems' complexity. We argue that the real issue is reducing complicatedness. This is an important

distinction. Complexity can be a desirable property of systems, provided that it is *architected complexity* that reduces *complicatedness*. Complexity and complicatedness are not synonyms. Complexity is an inherent property of systems, and complicatedness is a derived function of complexity. We introduce the idea of complicatedness of complex systems, present equations for each and show that they are separate and distinct properties. To make these ideas actionable, we present a design methodology to address complicatedness. We show examples and discuss how our equations reflect the fundamental behavior of complex systems and how our equations are consistent with our intuition and system design experience. We discuss validation experiments with global firms and address potential areas for further research. We close with a discussion of the implications for systems design engineers. As engineers, we believe that our strongest contributions are to the analysis, design and managerial practice of complex systems analysis and design.

Tang and Salminen [TAN 01] illustrated the difference between complexity and complicatedness. Relative to a manual transmission, a car's automatic transmission has more parts and more intricate linkages. It is more *complex*. For drivers, it is unquestionably less *complicated*, but for mechanics, which have to fix it, it is more *complicated*. This illustrates a fundamental fact about systems; *decision units* act on systems to manage their behavior. Complexity is an inherent property of systems. Complicatedness is a derived property that characterizes an execution unit's ability to manage a complex system. A system of complexity level, C_a , may present different degrees of complicatedness, K , to distinct execution units E and F ; $K_E = K_E(C_a) \neq K_F = K_F(C_a)$.

There is no research on complicatedness and complexity as distinct properties of systems. Research seems to cluster around engineering management and physical products. The focus is on modularization and interactions with a bias to linear systems and qualitative metrics. There are efforts on methodologies and tools, but theory, foundations and software have a demonstrably lesser presence. The work of Ferninand on software systems complexity is a happy exception [FER 93]. It is analytical, rigorous and elegant. The services and enterprise solutions are barely addressed. This is a serious omission given the high proportion of services in industrialized economies. Although layering of abstract systems and reintegration has a long history, the literature is skewed to decomposition rather than integration.

Overwhelmingly, the literature considers a system with a large number of elements as complex. Very few address the linkages among the elements, and

no one considers their bandwidth. All these factors are inherent characteristics of systems. Therefore, Tang and Salminen [TAN 01] argued that the number of elements, the number of *interactions* among them and the *bandwidth* of these interactions determine the complexity of the system. As any of these increase, we expect complexity to increase. For example, a system $\mathbf{N} = \{\mathbf{n}_i\}_{i=1,2,\dots,p}$ with binary interactions among the elements. Complexity, C_N , of this system does not exceed p^2 , we denote this by $C_N = O(p^2)$. System $\mathbf{M} = \{\mathbf{m}_j\}_{j=1,2,\dots,p}$ can have complexity $C_M = O(p^k)$ where $k > 2$. When $\mathbf{M}$ admits $\{\mathbf{m}_j\mathbf{x}\mathbf{m}_i\}_{j,i}$ and $\{\mathbf{m}_j\mathbf{x}\mathbf{m}_i\mathbf{x}\mathbf{m}_s\}_{j,i,s}$ interactions, $C_M = O(p^3)$. If $\mathbf{M}$ admits $\{\mathbf{m}_j\mathbf{x}\mathbf{m}_i\mathbf{x}\mathbf{m}_s\mathbf{x}\mathbf{m}_t\}_{j,i,s,t}$ interactions, $C_M = O(p^4)$. This characterization of complex systems admits systems with feedback loops of arbitrary nesting and depth, and high bandwidth interactions among system elements. Complexity is a monotonically increasing function as the size of the system size, number of interactions increases and bandwidth of interactions increase. In the limit, complexity $\rightarrow \infty$. We define complexity by $C = \mathbf{X}^n \mathbf{\Sigma}_b \mathbf{B}^b$

where $\mathbf{X}$ is an integer denoting the number of elements $\{\mathbf{x}_e\}_{e=1,\dots,p}$

n is the integer indicated in the relation $O(p^n)$

and $\mathbf{B}_1 = \sum_{ij} \lambda_{ij} \beta_{ij}$

λ_{ij} is the number of linkages between x_i and x_j

β_{ij} is the bandwidth of the linkages between x_i and x_j

$\mathbf{B}_2 = \sum_k \lambda_k^{ij} \beta_k^{ij}$

λ_k^{ij} is the number of linkages between x_k and (x_i, x_j)

β_k^{ij} is the bandwidth of the linkages between x_k and (x_i, x_j) and, in general,

$\mathbf{B}_n = \sum_n \lambda_p^{ijk\dots n-1} \beta_n^{ijk\dots n-1}$

$\lambda_n^{ijk\dots n-1}$ number of linkages among x_k and $(x_i, x_j), (x_i, x_j, x_k), \dots, (x_i, x_j, x_k, x_l, \dots, x_{n-1})$

$\beta_n^{ijk\dots n-1}$ linkage bandwidth among x_k and $(x_i, x_j), (x_i, x_j, x_k), \dots, (x_i, x_j, x_k, x_l, \dots, x_{n-1})$

$\mathbf{B}$ is a measure of the information capacity among the elements of the system. Note that the monotonicity properties are not violated. In Figure 19.2, an example is given.

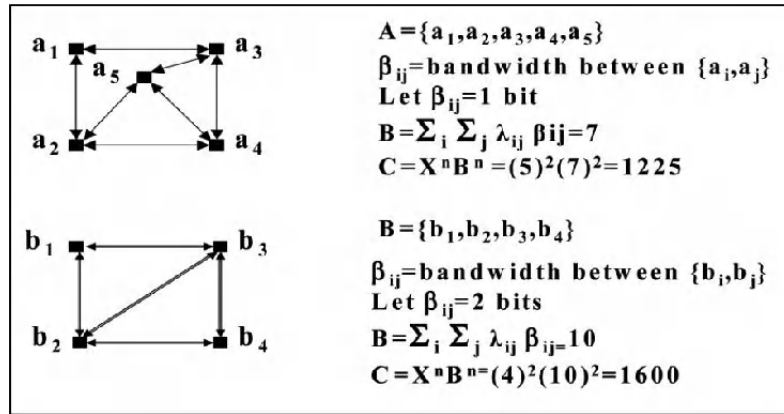


Figure 19.4. Complexity example of two systems

Complicatedness is the degree to which a decision unit for the system is able to manage the level of complexity presented by the system. The decision unit can be another system or a person. Complicatedness is a function of complexity, $K = K(C)$. Let us explore the properties that we expect from a complicatedness function. We expect that monotonicity of complexity is imposed on complicatedness, but do not expect that they are identical. Clearly at $C = 0$, $K = 0$. Consider K when $C \rightarrow \infty$. Intuitively, there is a level of complexity at which the decision unit can barely cope with the system. The system is becoming unmanageable. For example, most people can visualize a graph, $g = g(x, y)$ of $C_g = O(p^2)$, but it is harder for $h = h(x, y, z)$ with $C_h = O(p^3)$. Few can visualize a surface with four variables, although complexity has only reached $O(p^4)$. Consider equally incomprehensible systems A and B, where $C_A = O(p^{100})$ and $C_B = O(p^{100,000})$, respectively; $K_A \gtrsim K_B$ although $O(p^{100,000}) \gg O(p^{100})$. Therefore, when $C = 0$, $K = 0$ and when $C \rightarrow \infty$, $K \rightarrow K_{\max}$ asymptotically.

Accordingly, Tang and Salminen [TAN 01] systems are designed to operate and to be managed approximately at an optimal point of complexity, say C^* . For $C < C^*$, although the complexity increases, it is well within the interval of manageability. At $C = C^*$, the system complexity is optimal for the decision unit. For $C > C^*$, complexity is increasing and the decision unit can manage the system with decelerating effectiveness. Mathematically, $dK/dC > 0$ in the open interval $(0, \infty)$. At $C = C^*$, $dK/dC = 0$ and $d^2K/dC^2 = 0$. Complicatedness has reached an inflection point. So that for $C > C^*$, $d^2K/dC^2 < 0$, that is complicatedness is reaching saturation. The

decision unit's ability to manage complexity has reached diminishing returns. For $C < C^*$, $d^2K/dC^2 > 0$, complexity is growing faster than complicatedness. Because the logistic function is one of the simplest mathematical expressions that have all the above properties (Figure 19.3). We adopt it to express complicatedness. $K(C) = K_{\max}/(1+e^{-\alpha C})$

where e is the transcendental number $e = 3.2718\ 2818\ 284\dots$;

α is a constant specific to the decision unit;

C is the complexity of the system.

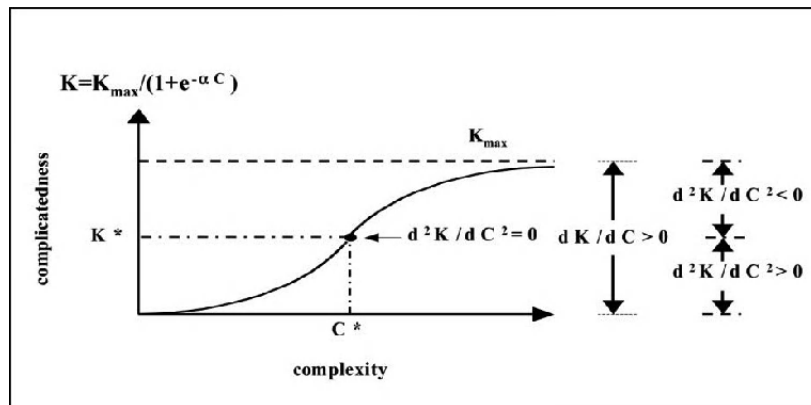


Figure 19.5. Complexity and complicatedness

Without loss of generality, we set $K_{\max} = 1$ to indicate the abject complicatedness. There are other functions, such as the Gompertz curve, Weibull distribution and log-reciprocal function, that can be used. The major differences are the location of the inflection point, the growth pattern before and after the inflection point, and the symmetry around the inflection point.

Earlier, we presented the automobile transmission as a complex system that is uncomplicated. Neural networks are more interesting as a systems engineering example. Typically, they are applied to situations where there are an intractable number of data points to analyze in order to set a course of action. To solve this difficulty, the neural network is layered, see Figure 19.6. The complexity is increased relative to the input vector. Many new elements, new interactions and their bandwidths have all increased the initial

complexity. But *architected complexity* has reduced an intractably complicated input vector to an output vector that now makes the system manageable. This approach has been proved effective for engineering paper machines [PIL]. This is a non-trivial example. The purchase price of a paper machine ranges around \$50 million and more. The mill generates about 10^9 data points, which are processed in real time by adaptive and distributed neural networks embedded in the machine. We experienced this case practically on a running paper machine in Finland and elsewhere to which we may revert at the end of this chapter, where we bring to the readers' attention the "live experiments".

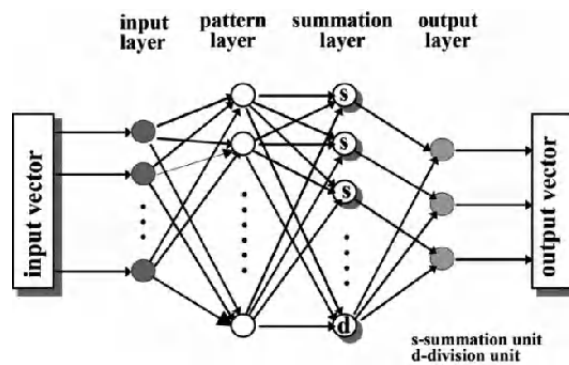


Figure 19.6. Use of neural network as architected complexity

The telecom infrastructure is one of the most largest systems in the world. On demand, it interoperates an immense array of networks, products and computers. The system complexity is enormous, yet we routinely make transcontinental telephone calls and download music and pictures from the Web. *Architected complexity* has made telecom networks manageable. Engineers created the OSI (Operating System Interface) reference model by partitioning the network functions into distinct layers. This architectural innovation creates, at each level, a distinct presentation of the network that is more abstract at each successive layer. Each layer presents to decision units a specific *system image* of the network that is vastly less complicated. Layering system images is a widely adopted doctrine in computers, for example programming languages. With the first computer, applications programming was very difficult. Programmers had to embed arcane hardware details into their algorithms. High-level languages were invented to present an abstract, but domain specific, system image for programming. A layer of software hid and encapsulated, transparently to the programmer, all machine specificities.

Architected complexity is a very effective complexity management strategy; it reduces complicatedness.

It suffices to present three examples. The typical video cassette recorder (VCR) control panel is a classic example of complex and complicated design. Another example is PC software or “bloatware”. So many application packages are functionally so extravagant that the average person can learn only a fraction of their functionality. Cellular phones are in danger of becoming examples of complex and complicated products.

There is good and bad cholesterol. Similarly, there are *architected* and *unarchitected complexity*. The former reduces complicatedness, whereas the latter does not. There are two important principles in *architected complexity*: partitioning the system into modules and reintegrating them while maintaining the systems integrity. Many decomposition schemes address the first principle; Karnaugh maps for digital circuits, Djysktra architectures for computers, design structure matrix for mechanical products [GHA 99], etc. They are effective tools, but when the decomposition creates a large number of new components and interactions, the result can become intolerably complicated and make reintegration impractical. Reintegration is less visible in research, although widely practiced by engineers.

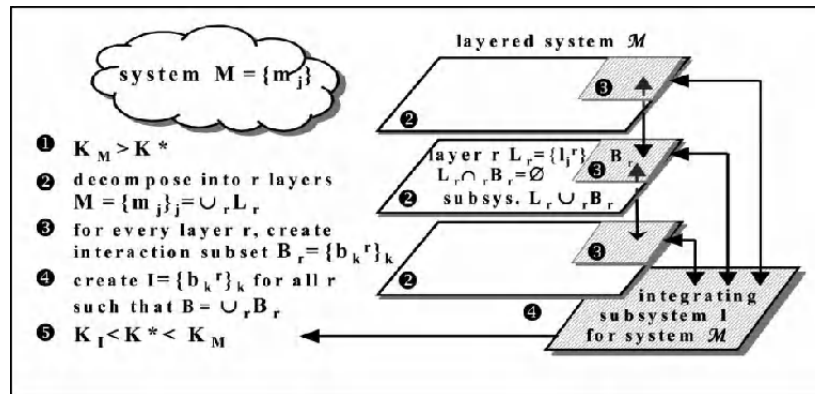


Figure 19.7. Architected complexity to reduce complicatedness

According to Tang and Salminen [TAN 01], the goal is to architect complexity so that M is transformed into $\mathcal{M}$, such that $K_{\mathcal{M}} < K_{M^*}$, although $C_M > C_{M^*}$. Partition M into layers, $L_r = \{l_j^r\}_j$ such that $M = \cup_r L_r$, that is all the

elements of $\mathbf{M}$ appear in some specific layer. Functional decomposition is an engineering application of this principle. For a paper mill, these can be the mechanical, control and process domains [HUB 88], or for a computer, the arithmetic unit, main memory, the I/O units, etc. Design the layers so that there are only intra-level interactions among the elements of a layer. Create $\mathbf{B}_r = \{\mathbf{b}_k^r\}_k$ for every layer $\mathbf{L}_r$, so that $\mathbf{L}_r \cap \mathbf{B}_r = \emptyset$. Design $\mathbf{B}$ so that only elements of $\mathbf{B}$ communicate with each other. $\mathbf{B} = \cup_r \mathbf{B}_r$ is $\mathbf{M}$'s communications subsystem. For $\mathbf{M}$, design a system integration unit $\mathbf{T} = \{\mathbf{t}_x\}_x$, which on one side interfaces with $\mathbf{B}$ and on the other side with the decision unit. Note that $\mathbf{T}$ presents the decision unit with an *image* of the system $\mathbf{M}$. This is a hallmark of a good architecture. Good design always presents a less complicated *system image* to a decision unit.

We have seen here that when mastering complexity versus complicatedness it seemingly is a challenging aspect, though very scientifically interesting. This part of the text gives you an insight about its nature and solution procedures when one enters into this area. It is time to turn the topic of methods and solutions for design, a product development scheme; earlier, we mentioned an original story in this area. When we plan to produce a faster product development scheme, we would imagine that there must be benefits to this phenomenon soon or at the end, when we are able to execute it into production, at a level before or later. These benefits, that we are now fashioning, have some impact on overall productivity and an alternative selection criteria to draw upon.

19.4. Benefits of fast product developments

When aiming at a higher product development and design speed, a company must, as a rule, change its traditional internal working methods. This will put a heavy strain on even the most positive of corporate climates. Furthermore, in such a case, the main contractor must coordinate its product development activities with its network of subcontractors.

The transfer from research to pragmatic design and manufacture is difficult when extensive entities and complex matters must be dealt with. There are phases during which the development work must be cyclically accelerated, alternating with phases for setting the sights for future times, for creating new visions. When creating breakthrough products, the development process itself should be developed so that the information and knowledge gathered through experience, can be documented and retained in the active

data storage. The essential part of the historical information on the various evolution versions of the product must be reusable to speed up the evolution process as much as possible. The modular structure of the multi-technical product is then extremely essential so that in the development work it is possible to concentrate the resources to the most important area.

The fast product development brings in at least two main benefits:

- When we get the product on to the market before the competitors come out with their own equivalent or better due to an early access benefit which, at best, gives a higher price and a larger market share (Figure 19.3).
- In time we develop new products faster than our competitors; we can start its R&D project later and still make a better success because the quicker products would be able to use the latest technology and fresher market information.

While launching a new product on the market, timing is often a key factor in whether or not the faster production process would make a better profit. A correctly timed introduction gives us a large market share right from the beginning and also helps us build up volume throughout the entire lifecycle of the product. Luckily, the speedier a product development project is, the less unforeseen problems are likely to occur during its course. The reason is that neither the market requirements nor the rival products can change much during a short development process. Consequently with a higher speed, the predictability improves. However, when developing the new products and building up timetables, one must always make provision for redesign, because it is inherent in this work that one must face the “new” and therefore be prepared for some degree of uncertainty. To understand the problems while building up a timetable, we should mainly deal with certain special characteristics of product development.

By developing its technology and product strategy, a company may significantly improve the proper prioritization of its R&D projects, and thus the efficiency of the entire production design process [CAR 93]. Speeding up of product development may be examined in two ways:

- developing the product development process (that is recognizing the value-added functions, flexible team work and continuous improvement);
- developing the product concept (i.e. especially modular concept).

The combination of these two is very important for the future. It gives us a possibility to use the sourcing of technology in a controlled manner. The

division of labor can be developed more speedily within the company and when working with subcontractors. This division offers a natural cause to speed up the product development scheme when part of the responsibility for product development could also be shared with the subcontractors.

A mechatronic product would be described according to Figure 19.8, when the interfaces between different sub-areas are emphasized [SAL 94]. Dividing the product with the help of interfaces is also fundamental to the modularization procedure.

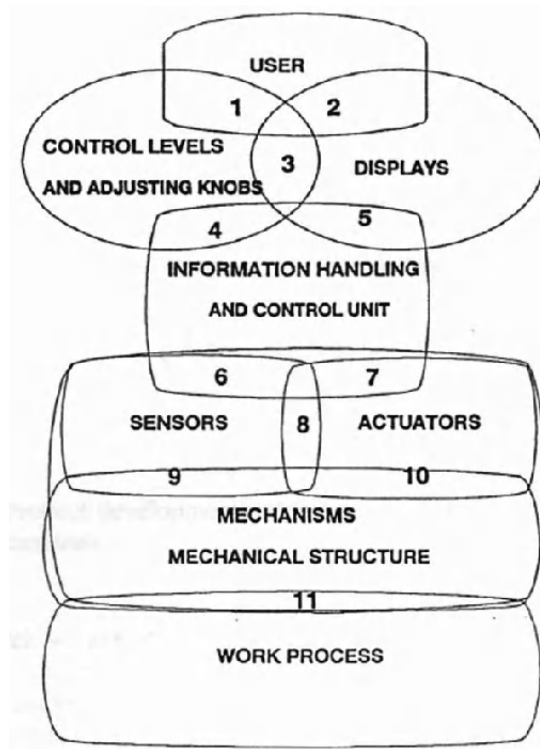


Figure 19.8. *Model of the major interfaces of a mechatronic product*

In Figure 19.8, interfaces 1 and 2 form the actual man–machine interface. It should be as user-friendly as possible. Sometimes these two interfaces are combined, see at interface 3. Interfaces 4–7 represent the distribution of information handling, when intelligence is integrated to controllers, displays,

sensors, actuators and mechanisms. On the other hand, interfaces 8, 9 and 10 characterize the integration of sensor functions and actuators to the mechanisms. However, the control electronics today are often integrated into the mechanisms compared with the 19th Century setups.

19.5. Nature of product development process

Twiss pragmatically synthesized the nature in saying that research is the antithesis of certainty. All that the formal scheduling systems can do is that they may raise the ratio of objective and subjective criteria in planning and decision-making [TWI 80]. The real innovative ability of a company is based on the information and know-how accumulated or generated in the company. For this reason, the innovative ability must be developed over a longer period of time. The usability and accessibility of the accumulating information are also key factors when we are aiming at a faster implementation of a product design project. Information gathering must, therefore, be managed in a purposeful way, bearing the needs of future product design projects in mind.

In research, we are moving on a partly unknown territory and hence, must be prepared for unexpected surprises, even unpleasant ones, in the planning and steering of research activities. Furthermore, there is a particular problem with scheduling and building up of timetables in R&D, that are not found in routine projects. This is a problem of related interdependencies (cross-coupling), where coupling of tasks is not serial or parallel, but a more complex one (Figure 19.9). Finding a solution for such awkwardly coupled tasks requires iteration. How much time this takes depends on two factors: the time taken by one cycle of iteration and the speed of convergence. If we must include research work in such an iteration cycle, we may have to waste a lot of time. Therefore, it is important to complete all search and exploration early enough to allow us to perform the actual product design project as fast as possible.

Practical product design projects and numerous studies show that there is a somewhat fuzzy phase of preliminary studies and surveys at the beginning of a design process. Too often, these studies are conducted as part-time jobs, between other tasks. Furthermore, they often lack an objective and a timetable. The working methods, too, may be erratic and somewhat spontaneous. All this wastes the precious calendar time reserved for the project [REI 91]. An extensive study [COO 90] conducted in the United States states that the product development projects leading to successful businesses are different from the rest in that distinctly more work was put in at the early stages of the project. We should note particularly that the total

amount of work at these early stages is very small, and, consequently, the costs are fairly low, at most a few percent of the total expense of the entire product development project.

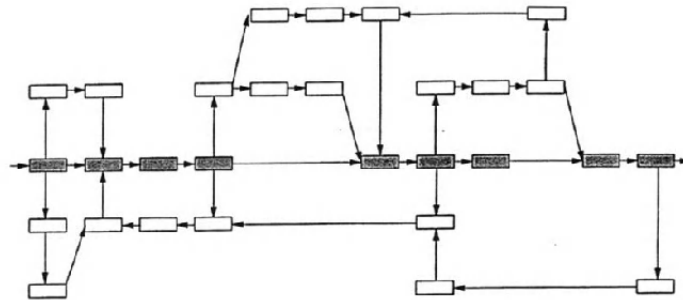


Figure 19.9. *Product development process, critical path and feedback loops caused by iteration*

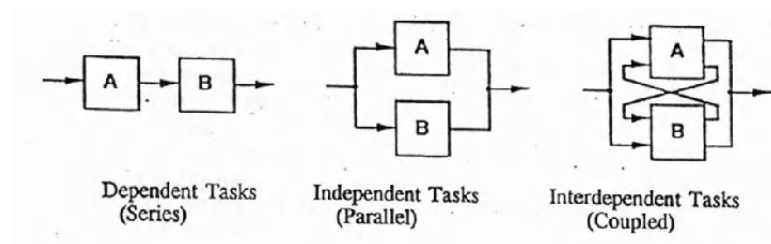


Figure 19.10. *Three possible sequences for two design tasks [EPP 91]*

When building up a concept, a critical, most problematic point or area is circumvented too easily. It is difficult to recognize critical subprocesses. Usually, these are the ones where the widest technology leaps are made and also where the greatest risk is taken. The most sensitive points on the critical path should be solved first if we want to shorten R&D time and accelerate the process. When increasing speed, the degree of integration on the critical path must be defined precisely. When deciding which functions should be developed parallel to which others, special attention must be paid to the weak links, i.e. the manufacture of the prototype, testing and the utilization of the results. With a little extra work input at the beginning of a product design project, it is possible to accomplish a lot, both shortening the time required and guaranteeing the business success. This stage provides one of the

cheapest and best chances of speeding up a product development project [COO 90]. The projects studied show that about half of the product design time is lost before the nomination of the project group [SMI].

19.6. Planning the timetable of a product design project

The same scheduling principles apply to a product design project as to engineering projects in general. The design project must be divided into tasks, and the interdependencies between these must be distinguished. The relations between the tasks may be of three types as explained earlier, i.e. parallel, serial or coupled. A genuine parallel relation between the tasks makes the scheduling easy because these tasks may actually be done in parallel and without interdependencies. Even the serially connected tasks are easy in that the tasks may be absolved independent of each other as long as they are performed in the right order. On the other hand, the coupled tasks are difficult from the viewpoint of making up a timetable and keeping it because solutions to these problems must often be sought by some kind of iteration process. Therefore, they often require special measures by the project manager. For instance, if such coupled tasks are performed within different organizations, there is a risk of the parties only waiting for other partners to give the start-up information, with no progress in the actual project. Hence, it is important that such issues between the tasks are recognized at the project planning stage.

The goals of product development and product design should be business-oriented. Although the product should be excellent from the viewpoint of the selected customers, the product designer must not merely optimize the functional characteristics of the machine; he should mainly aim at optimizing our business. It follows from this principle that both the requirements of the customer and the business requirements of company should be taken into account when setting goals. From the viewpoint of speedy product design, it is important to keep the goal-setting as clear and unambiguous as possible. It should contain the following objectives:

- operational objectives;
- quality objectives;
- production objectives;
- marketing objectives;
- maintenance objectives;

- economic objectives;
- objectives of logistics;
- objectives of interest groups;
- the company's objectives;
- timetable objectives.

These objectives should be expressed as clearly as possible, preferably quantitatively, even in the case of quality objectives. In setting the goals for product design, we must pay special attention to the degree of change that must be made in the product to secure its competitiveness. Whether or not a conceptual change is made in the product generally makes a great difference in the speed at which the project is carried out [HOL 90]. Industrial products vary greatly in terms of design speed. From the viewpoint of customer flexibility, they may be divided into the following categories:

- serial products, delivered to all customers, identical in form and function;
- products assembled from standard parts, according to the customer's order;
- products designed and manufactured fully according to the customer's order (tailor-made);
- various combinations of the above; most investment goods fall into this category.

Factors affecting the economy of the development of products and services are, for example, the time at which they are launched on the market, their characteristics and the technology used to develop them.

Figures 19.8 and 19.9 schematize the interfaces of a mechatronic product and product development process, respectively. Figures 19.11, 19.12(a) and 19.12(b) schematize the implementation of development and design projects based on either a static or a dynamic product development concept. See Figure 19.11.

The design of a product based on a static concept would be accelerated and developed also by automating the design process. On the other hand, a product based on a dynamic concept mostly requires more iteration and development time.



Figure 19.11. Product design based on a static product concept

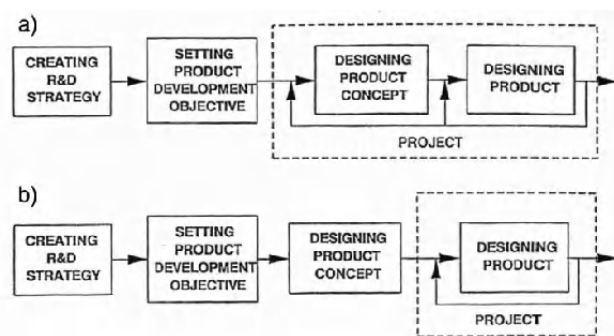


Figure 19.12. Product development based on a dynamic concept [SAL 94]: a) fully dynamic concept with strong interaction between the concept and its elements and b) semi-dynamic product concepts with interactions between the concept and its elements, weak enough to allow changes without extensive iteration.

19.7. Designing the product concept

Creating a product concept is in many respects an extremely important phase of the design process. It largely determines the flexibility of the product entity in meeting the special requirements of each individual customer. The product concept also decides how fast, reliably and economically we will be able to serve the customer after he has given us his order. The choice of a product concept often has a very strong impact on the speed of the actual product development.

19.8. Enhancing conceptual design

The objective of a conceptual design should be fewer and simpler interfaces. The number of technical elements in a modern product may be high. Consequently, the number of interactions between the elements is very

high as well. For instance, in the system in Figure 19.7 with $3 \times n$ elements, the theoretical number of interfaces between the elements is $3 \times n/2 (3 \times n - 1)$. Although in actual systems, interactions do not occur between all the elements, the risk of making the concept too complicated at the system design phase remains. The reason usually is that not enough attention is paid to the interfaces and their number (see Figure 19.13).

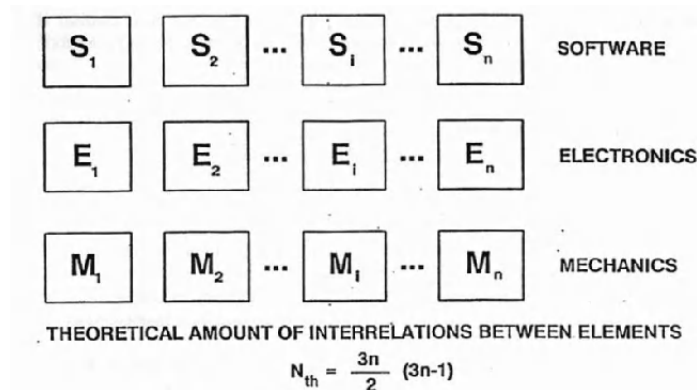


Figure 19.13. Theoretical number of interfaces between elements in mechatronic technical systems with $3 \times n$ elements [SAL 94]

In speeding up and scheduling a product development project, key questions related to conceptual design are the following [SAL 94]:

- What functions should be included in the product?
- What are the inter-relationships between the functions?
- What functions will be performed and by which part of the machine or module?
- What are the performance data of the product modules and what are the safety margins of the design?
- What are the interfaces between the elements? These may be both geometric-physical contacts and signal transmission methods as well as protocols related to the flow of information.
- How will the reliability objectives of the overall product be divided among the various modules?

Although we are faced with asking and finding solutions for such essential questions at the phase of designing the product concept, it is often largely neglected; many of these questions may never even be subjected to explicit consideration. This involves considerable risks for the future of the product. Even from the viewpoint of fast product development, the decisions made at this phase are very far-reaching. Decisions related to the product concept may essentially slow down the product development project. The implementation speed of such a project is largely decided by the choice of the product concept and especially the module construction linked with the design of the interfaces. What is perhaps worst is that these decisions also affect the speed at which this product can be updated during subsequent product development cycles. The determination of the product concept may predestine the development speed of future product generations as a slow one.

It thus follows from the above that we, when designing a product concept and making choices affecting it, must necessarily pay attention to the effect of the concept on the speed of product development. Below, we take a look at the most important factors affecting the development speed through the choice of product concept [SAL 94]:

- 1) An appropriate module construction may promote the progress of parallel design teams working on different projects at the same time.
- 2) By using standard components and modules, we may concentrate our efforts on developing the product characteristics that are the most important ones for the customer and for our own competitiveness.
- 3) By the use of modules, we may also promote subcontracting, both in product development and manufacture.
- 4) The appropriate division of product development risks among different modules gives us a better chance to control the risks involved in both the product itself and in the scheduling of the product development project.
- 5) If we divide the product into modules in a way that allows the fastest-developing technologies to be contained in their own modules, we will be able to update the product design quickly without having to change too many details.
- 6) As the product concept, we should choose a solution that is as clear and simple in its interconnections as possible.
- 7) The concept should be insensitive to minor changes in the interconnections in the adjacent modules.

These principles and possibilities are of great importance and should be taken seriously because they give us extensive chances to make both the product design and, in part, the production more efficient. In the conceptual design of a machine, we must make several important decisions concerning the structure of the machine. This structure may be looked at on several levels and from the viewpoint of several different interactions. The key interactions are [HUB 88]:

- interaction in the functional structure;
- interaction between the functional organs;
- interaction between the constructional elements.

The constructional elements may be divided into three categories as follows [HUB 88]:

- internal constructional elements;
- border elements forming a link between the internal subsystems of the total system;
- boundary elements located on the outer surfaces of the technical system.

Among these, the second category, the border elements, is important when we consider the constructional elements of a machine that have an effect on the speed of product development.

19.9. Interaction between the parts of the machine

The interfaces between the parts and modules of the product and the complexity of the interactions across them are very important for the speed of product development. It is in the interfaces between the machine parts that the functional, physical and constructional interaction happens, making the intended function of the whole product possible. Essential parts of this interaction are the information, energy and material flows⁶. All these flows have an effect not only on the function of the product but also on the process of developing it and, in particular, on the complexity and speed of product development. Another extremely important factor in the interaction of the machine parts is their geometry.

⁶ This is mentioned earlier in relation to Figure 19.8.

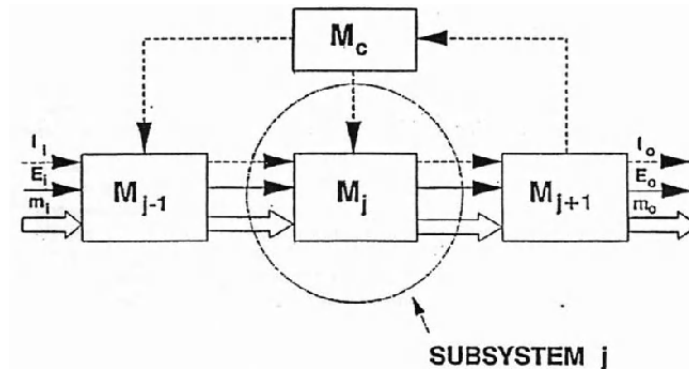


Figure 19.14. Development of a module with consideration of the information, energy and material flows between it and the adjacent modules (a product consisting of serially coupled modules)

The interaction between the machine parts can be looked at from many viewpoints. Below, we try to find those features of interaction that have an effect on the speed of product development. A particularly important question is “how strongly a change in the structure of an adjacent element (machine part or module) affects the elements connected to it and how large a volume of design work is made necessary by this change to modify these elements according to the new requirements?”

19.10. Effect of the strength of interaction between product parts and development speed

The interaction between modules to be connected to each other and the development speed has an essential impact on the process [SAL 94]. It is divided into three categories: weak, semi-strong and strong. The interaction suggested here may be functional, physical, chemical, geometrical and related to the information content that are in transition. At most, loss of product development time is caused by the physical (and chemical), geometrical and informational (control, information handling) aspects. Components and modules with weak interaction are characterized by the fact that they hardly cause any delays in the convergence of the product development process. Components with strong interaction, on the other hand, slow down the convergence in product design and development, which leads to the lengthening of the entire product development process when the

elements in question happen to be on the critical path of the design process. This discussion is briefly summarized in Tables 19.1–19.3.

Machine parts with weak interaction	Machine parts with semi-strong interaction	Machine parts with strong interaction
Wireless signal transmission methods Non-contacting sensors Ropes Electrical cables Pipes and tubes Chains Belts and other flexible bands and strips	Flexible shafts Articulated shaft lockings (universal joints, Hooke's coupling, etc.) Springs (coil spring, rubber and plastics, leaves etc.) Actuators (hydraulic and pneumatic)	Machine body/frame Shafts & spindles Bearings, Screws Welded, flexible, solder and glued Joints Electromagnetic power transmission parts (when a good efficiency is the objective)

Table 19.1. *Classification of typical machine parts according to strength of geometrical interaction*

Functions with weak interaction	Functions with semi-strong interaction	Functions with strong interaction
On/off signal transmission Signal transmission in slow measuring Transmission of sensor signals of a scalar quantity Slow transmission of signal	Inter-module signal transmission in a modular control program Rapid transmission of simple signals	Rapid measuring and control Navigation in a changing environment Graphics, pattern recognition Embedded control (interface between electronics and software modules)

Table 19.2. *Classification of informational functions according to strength of interaction*

The interaction between the informational functions and the geometrical interaction between machine parts are the areas of product development that usually require most design work (Table 19.3).

Informational interaction Geometrical Interaction	Weak	Strong
Weak	On/off signal (overpressure sensor) Warning light cable	Gear wheel
Strong	Embedded control Image handling and transmission Navigation	Electrohydraulic actuator with control Electromagnetic bearing with control Electropneumatic actuator with control

Table 19.3. *Interaction between informational functions on one hand and geometrical interaction between other machine parts on the other hand in various parts usually found in a machine system*

The stronger the interaction is, the harder it is to work out the design. This means that the design work gets laborious and slows down as we move down and to the right in the field of Table 19.3. The interaction between the information system and the mechanical parts is today gaining more and more importance. This is especially true of the interaction between the control and adjustment algorithms of mechatronic systems and the mechanical parts of the machine. This interaction may, at its most difficult, be quite strong and complicated.

So far, we came up with terrifying problems in identifying and tracking the idea to a physical mechatronic product, and it is possible now to lead you in configuring and isolating mechatronic products. Mechanical engineers, who think of a product as a complete “*end-to-end one*” and stand-alone system, need not necessarily specify its surrounding features that would need backup and updating processes, including continuous maintenance through spares and other intimated agendas. They consistently assume that a physical product has other agendas that are included in the specification and do not repeat them. We simply look at a physical product and perhaps its other agendas such as after-sales service or other lifecycle coverage potentials. Instead of going into details, we define what the product is and what service it gives and then combine them to understand the phenomenon much better than before.

19.11. Definition of product and service

The product and service need to be defined. An engineer assumes when he produces a technical product, which has internally a maintaining service domain. On the other hand, a service provider assumes that the concept consistently offers not only the product but also services. Design engineers and product users are misunderstanding; or assuming, what is a product and what is a service. It is like the fundamental law (Coulomb's law⁷), which states that the electric force of attraction or repulsion between two point charges is directly proportional to the product of magnitude of each charge and inversely proportional to the square of the distance between them. The force also depends on the permittivity of the medium in which the charges are placed. If Q_1 and Q_2 are the points of charges at distance d apart, the force is:

$$F = \frac{1}{4\pi\epsilon} \frac{Q_1 Q_2}{d^2}$$

where ϵ is the permittivity of the medium.

The force F is attractive for opposite charges and repulsive for the same charges. In this context, it is addressed that product and service are scientifically attracted. They are treated or understood to be one and the same. One may interpret it either way because of the purpose paradigm. It is further defined so as to understand the phenomenon in the form of a molecular structure, see Figure 19.15.

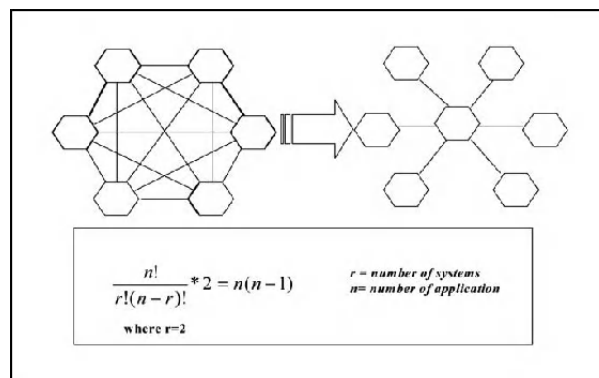


Figure 19.15. *Product centric versus service centric*

⁷ Dictionary of Science, University of Cambridge, UK, 1998.

It is shown in the above figure that product and service are two different entities that are not well defined but can be seen when distilled. Let us look at the same phenomenon from a different angle. In physics, the rate of change of the heat Δh of a process with temperature, v , carried out at constant pressure, is given by:

$$(\sigma \Delta h / \sigma v) v = \Delta (\sigma h / \sigma v) v = \Delta C_p t,$$

where ΔC_p is the change in the heat capacity at a constant pressure for the same process.

Similarly, for a process carried out at a constant volume:

$$(\sigma \Delta h / \sigma v) v = \Delta (\sigma U / \sigma v) v = \Delta C_v.$$

Product and service platform, when optimized, give opportunities to align the patterns from one perspective to another. Requirement and dependency threads the guidelines to form metadata for product data management (PDM).

Modern complex human-machine systems are truly mechatronic in nature, such as the control room in nuclear power plants, the autopilot or pilot-aid-systems in the cockpit or aircraft, and a paper machine in generic terms. Lin *et al.* [LIN 01] identified and explained through their experiments that the parts displayed in physical laws or effects in the ecological interface design (EID) framework appear problematic. When the mental workload of a human being is higher; compared with his self-assessment, the complexity increases that threat of the complicatedness. We defined previously that the entire product coincides with a service manifesto, which is encountered as a separate “thing⁸” and has no effect on design and interoperability terms. We will proceed further to express the previous discussions on mechatronic design methods and isolate mechatronics as a physical product and shed new light into the services. The use of industrial robots in manufacturing plants has facilitated the massive production of goods with high quality and low costs. This chapter introduces some of the basic ideas used in our previous discussions through a couple of case studies; they are classified as mainframe mechatronic products but also include the new phenomenon known as the service.

⁸ The term “thing” here is meant as an axiomatic in its behavior, or say designers or creators are of the opinion that this is not important although the product itself is the matter.

19.12. The case studies

We will introduce a couple of cases studies, as mentioned in section 19.11, that are focused on a *paper machine* and addressed to a *model-based implementation case* based on *provisioning of live data* to a portal provider company. Pulp and papermaking involve numerous steps, many of which rely on chemical activities. Strategies to solve elementary schemes in the wet-end process of a paper or paperboard machine are of a vital interest within the global paper industry. Figure 19.16 shows a paper machine with control points and interaction needs. The papermaking is a real-time, stochastic and an intelligent learning process.

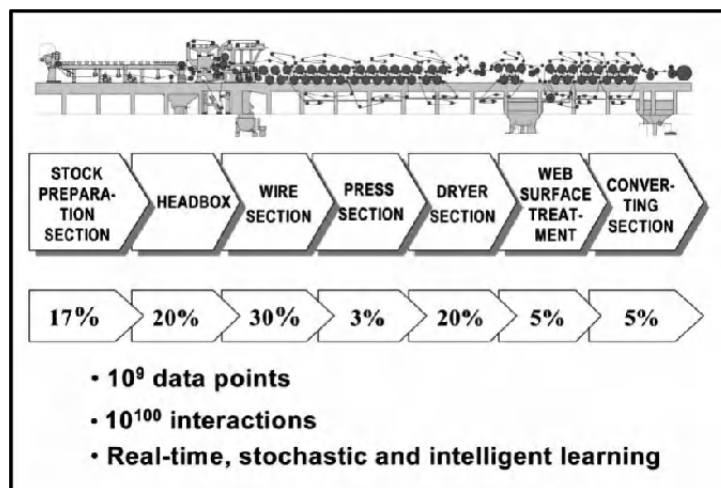


Figure 19.16. Paper machine concept

All the nonlinearity that appears in the chemical process of the wet-end (e.g. short circulation and headbox) of the paper machine is difficult to include in a global model. Machine wet-end must be decentralized and managed in small modules. The product design plays a significant role in the iterative and complex engineering process. Decisions made at the engineering design stage have a considerable impact on the product's lifecycle costs. With our real world, hands-on training, we learn to use new technology right the first time rather than waste time with costly trial and error. Today's technology advances at a relentless pace, while the "implementation gap" continues to widen.

The proportional, integral and derivative (PID) controller has been in use for over a century in various forms. In a typical PID controller, these elements are driven by a combination of the system command and the feedback signal from the object that is being controlled in a plant (or say paper mill). Figure 19.17 is a block diagram of a basic PID controller. In this case, the derivative element is being driven only from plant feedback. The plant feedback is subtracted from the command signal to generate an error. This error signal drives the proportional and integral elements. The resulting signals are added together and used to drive the plant. The resulting signals are added together and used to drive the plant.

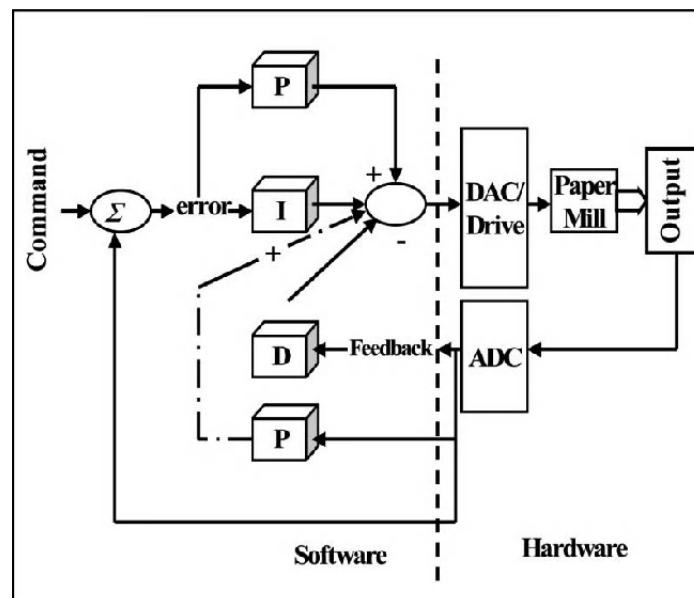


Figure 19.17. Basic PID controller scheme

A complex system communication is a collection of nodes and links that communicate by defined sets of formats and protocols. Within the network, there are usually three layers: transmission, switching and service. The transmission layer consists of systems, for example cables, radio links and their related technical equipment. The switching layer consists of switching nodes with generic and application software and data. The service layer, distributed among the switching network elements, consists of special hardware and their application software and data. This concept applies to

robotic systems or complex generic product deployment systems in the processing environment.

19.13. Networking systems and learning mechanism

Digital integrated services based on asynchronous transfer modes is the suitable technology for transmitting information from any media spectrum and applications. A cell-based, high-bandwidth, low-delay switching and multiplexing technology is designed to deliver a variety of high-speed digital communication services. These services include local area network interconnection, imaging, and multimedia applications as well as video distribution, video telephoning and other applications. The learning algorithms we executed to a robotic concept are of asynchronous form because the recurrence of cells or movements containing information from any degree of freedom is not necessarily periodic in nature [KOH 89]. Therefore, they can handle both connection-oriented and connectionless traffic with adaptation layers and operate at either a constant bit rate or a variable bit rate connection. Each module or cell sent into the network contains addressing information that establishes a virtual connection from the source to the destination. All are then transferred in sequence over this virtual connection.

Papermaking (Figure 19.16) is modeled onto subsystems that have a system of damping elements, and the equivalent damper can be shown using the same logic and similar steps. As is, in the event of a robotic structure, the number of degrees of freedom of a dynamic entity, such as papermaking, is defined as the number of independent generalized coordinates that specify the configuration of the system. Generalized coordinates need not be restricted only to the actual coordinates of position.

The position coordinates are physical coordinates. On the other hand, a generalized coordinate could be anything, e.g. positional coordinate, translational displacement, rotational displacement, pressure, voltage, fiber, chemical component or current. Generalized coordinates need not be of the same type. The generalized mechanical configuration can be a mixed set of translational displacement and rotational displacements. The physical layer has two sublayers: transmission convergence and physical medium-dependent. The physical medium-dependent sublayer interfaces with the actual physical medium and passes the recovered information (bit stream) to the transmission convergence sublayer. This sublayer extracts and inserts the module cell with the synchronous digital hierarchy time-division multiplexed

frame and then passes them to (and from) the basic model medium layer. The model medium layer performs multiplexing, switching and controlling actions based on information in the model medium cell header. This is what we experimented with in the wet-end configuration of a liquid paperboard-making machine environment. Before we switch to discussing the experiments, we address the non-holonomic constraints involved in the framework. There are two types of constraints: holonomic and non-holonomic constraints. Systems with holonomic constraints are relatively easy to deal with, whereas those with non-holonomic constraints are more difficult. Fortunately, many engineering systems that we often encounter contain holonomic constraints. For papermaking systems, the number of degrees of freedom is the same as the number of independent generalized coordinates, which is also the same as the number of independent differential equations. Holonomic constraints are equations expressed in terms of the coordinates and time. The classical wandering effect defined by Pillai [PIL] is an excellent example of holonomic constraints.

We experimented on a mechanical system that requires two independent coordinates to specify its configuration; this is called a two-degrees-of-freedom system in the papermaking process. The two-degrees-of-freedom is a special class of multiple-degrees-of-freedom systems. We express the equations without the standard second-order matrix as:

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{Bmatrix} = \begin{Bmatrix} x_1 \\ x_2 \\ \dot{x}_1 \\ x_2 \end{Bmatrix}, u_1 = f_1(t), u_2 = f_2(t), y = x_1 \quad [19.1]$$

Here, the equation contains the mathematical body and physical entity, where the stated equation is a set of four first-order differential equations, where many intelligent people prefer to apply Newton's second law for the mass m_1 and m_2 , respectively,

$$\rightarrow \sum F_x = ma_{cx} \rightarrow \quad [19.2]$$

$$\begin{aligned} f_1(t) - k_1 x_1 + c(x_2 - x_1) + (x_2 - x_1) &= m_1 \ddot{x}_1 \\ f_2(t) - c(x_2 - x_1) - k_2(x_2 - x_1) &= m_2 \ddot{x}_2 \end{aligned} \quad [19.3]$$

A check on the accuracy of the differential equations of dynamic systems is always useful. All the elements on the main diagonals are non-negative (either positive or zero). All the mass, damping and stiffness matrices are symmetrical (with respect to the main diagonals). The off-diagonal elements of both the damping and stiffness matrices are non-positive (either negative or zero), and the off-diagonal elements of the mass matrix are non-negative. A pure system is defined as strictly of one type, e.g. mechanical systems with purely translational or rotational motions. Liquid or fiber mix level systems are also dynamic and pure systems in this context.

With those dynamic entities, we have experimented with a non-holonomic colloidal search in the wet-end of a liquid paperboard-making machine. A robotic function was invented. We had installed seven high-speed digital cameras directed towards certain activities. Those activities are based on control parameters, chemical reactions and time. The time domain is based on the action initiated, action processed and action terminated or continued to the next stage, which is detected through some rules. Those rules are configured and adapted through fuzzy interpretations. The configurations are based on model-based domains. The adaptation layer configures the module-dependent actions. The model passes protocol data units to and accepts them from higher layers. Protocol data units may differ in variable or fixed length from the model length. The physical layer corresponds to the defined layer 1:3, and many are subconfigured protocols. The subconfigured protocols at the module configuration survive as an open system interconnected model, while the modular layer and adaptation layer correspond to parts of the open system interconnected model. The physical model consists of two logical sublayers: the physical medium-dependent and the transmission convergence. This is better understood by calling it the physical layer of the paper machine headbox. The headbox itself contains several sublayers. Those layers are comfortably operating in this case as independent model-based modules or sub-sub-layers. The transmission convergence layer functions and interprets the interfaces between the layer submissions. It also provides bit transmission capability, including transfers, alignment, line coding and electrical-optical conversions. The flow information between fiber bond including the surface value changes are transmitted and received over a physical medium. The transmission frame adaptation function performs the actions

necessary to structure the model flow according to the payload structure of the transmission frame, or conveying direction, to extract the module flow out of the transaction frame (receive direction).

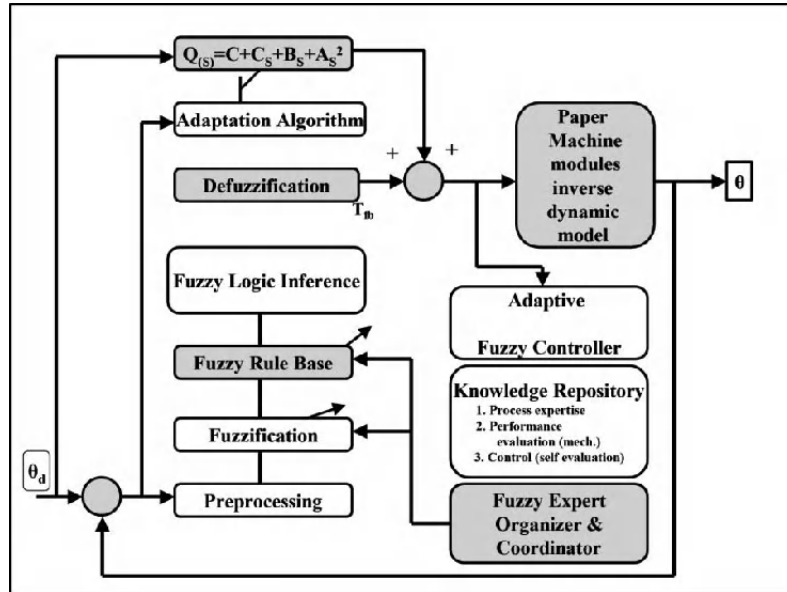


Figure 19.18. Adaptive fuzzy logic control scheme [PIL]

In the adaptation of network- and rule-based fuzzy logic [PRI 98] also, fuzzy modeling has been used in constructing a model for the papermaking and process, which equally accesses the referendum from the combination of the mechanical, process components with its control strategy. Figure 19.19 shows an approach for adapting a control strategy in the decentralized system as discussed earlier. The system is considered for all the players in the sequence where mechanical, process and control components fit neatly. This function is one of the several adaptive fuzzy controllers. $Q_{(s)}$ is the module system at any given moment. The subscript (s) indicates frequency domain. In the time domain, it will be:

$$Q_{(s)} = C + C_{(s)} + B_{(s)} + A_{(s)}^2 \quad [19.4]$$

where θ is a vector of control variables; A, B and C are coefficient matrices.

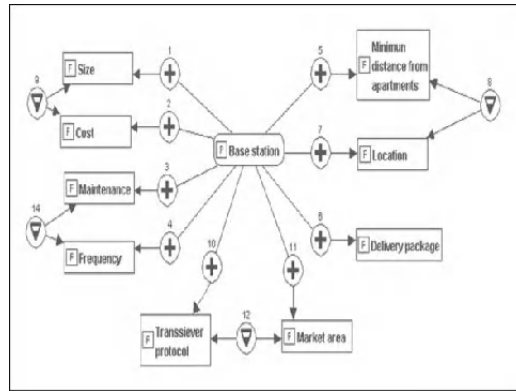


Figure 19.19. *The dependency setups of a telecom portal (a screenshot)*

We have seen that the complexity of a system like the one experimented upon is difficult to detect or converge. Salminen and Pillai [SAL] have defined the complexity for the purposes as follows:

Any system comprising of the varieties of interdependent and compound netting or parenting suffix, we call complex system or systems. A complex system, as we have defined, has many activities surroundings, which are not linear members of the functions. It performs as linear state functions. This linearity shifts, when process non-stability appears, due to gray-areas in the netting. This gray-area evokes the process wandering. This phenomenon we call nonlinear dynamics. This nonlinear syndrome is seen in human organizations, man-machine-interfaces, partnership setups and breathing. These systems are physical cell bodies that acquire, store and utilize knowledge. The connection weight is adjusted to achieve a desired output. The process is embedded in such a manner that it computes the derivative of the energy function with respect to the connection weight as an increment of the connection weight. This way, the derivative determines the rule of changing connection weights in order to minimize the descent of the participating energy function along the gradient. Therefore, we weigh here the feedback-error-learning and not linearizing the complex system. Obviously, also the product development with tools and skills is a complex system, which needs adaptive interpretations.

This discussion is not meant to offer any software solutions but to interpret the mismatch of engineering disciplines. We have a number of cases where syntax is present, but purely missing the ontology of the concept. We suppose that the engineering disciplines need to grasp a better understanding of the engineering ontology of all related or non-related entities for this purpose. Here, we introduced a case study using a computational intelligence technique, fuzzy logic and a learning network system for solving complex problems of papermaking process. There are many similarities in applying this hybrid methodology for the purposes of robotics. We could say now that the experience shows that the same approach could be used in most of the complex nonlinear, systems.

19.14. Model-based methodology: an implemented case

We conduct a case study based on the above framework. The study executed with a telecom portal provider [SAL 03]. We have dynamically integrated the product and service concept into an industrial reality. We used also the above mentioned software tool. We modeled the portal-provisioning live data, while acquiring at random, a number of unspecified customer requirements and its new features. The following figures would explain the use of methodology and the software tool.

Figure 19.19 (an actual computer screenshot) indicates the dependency of dismantled provision automation systems. The dependency; is further expanded with performance rules. The methodology of rule creation categorized is shown Figure 19.20.

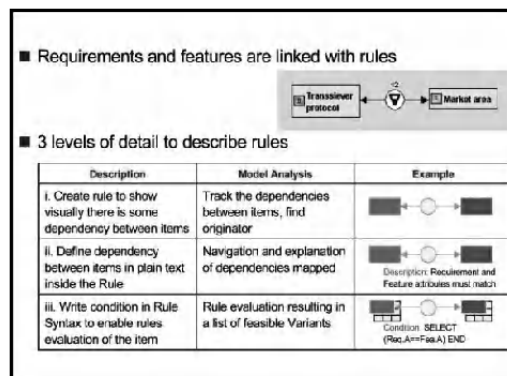


Figure 19.20. Rule creation system

While considering the lifecycle Challenge Management [SAL 03], LcC_{Mgt} of a product and service in this case is invisible and faster than the solid entity. Here as a research team, we had a problem in justifying this concept in general. However, Lifecycle is very short and fully packed with high technology. Here, the challenge makes sense. Therefore, we made an archive (Library) of requirement and used trajectory for tracking to build the “Backbone”. The software tool is then applied to interpret and view the “Backbone”. It is further scanned to pick up the required information from the archive to produce three-dimensional (3D) figures as to how a document or component is published or released collaboratively, “looks-alive”, before the decision portfolio. All these steps are shown in Figures 19.21 and 19.22. While creating methodology, we scheduled an adaptive learning algorithm.

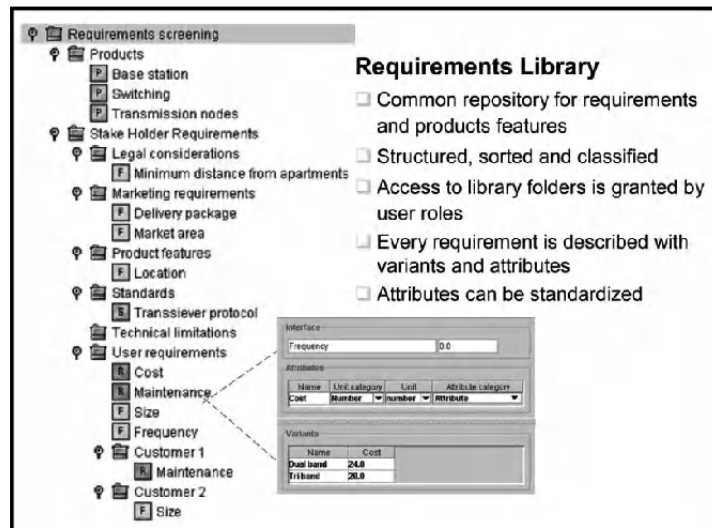


Figure 19.21. The use of requirements library

Integrating products and services into the new framework attracts the industrial needs in fast-growing technological areas. In the context of knowledge sharing, we use the term “ontology” to mean a specification of a conceptualization. That is, ontology is a description of the concepts and relationships that can exist for an “agent” or a community of agents. When knowledge of a domain is represented in a declarative formalism, the set of objects (e.g. service and product) that can be represented is called the universe of discourse.

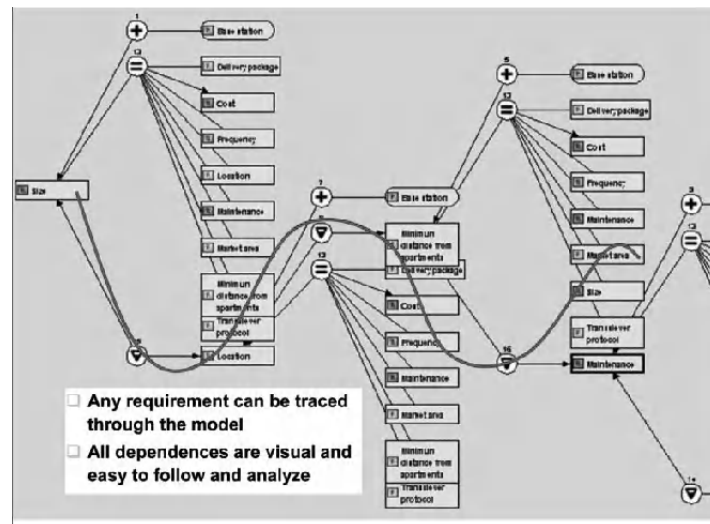


Figure 19.22. *Tracking from the trajectory*

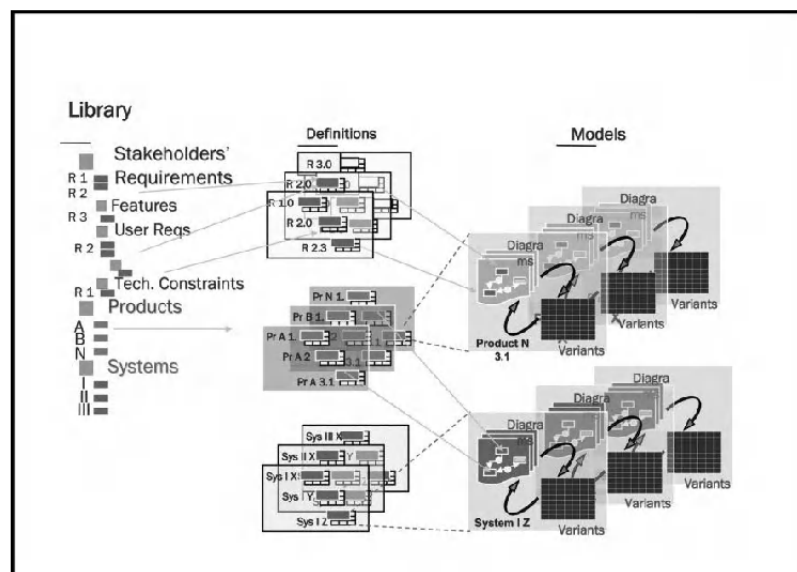


Figure 19.23. *Summary of the methodology used*

On the practical side, we automated the workload of a telecom provisioning system. The search engine and browser robot were created on a semantic infrastructure. The piloting results calculated are shown in Table 19.4 as an actual representation of the telecom portal. This project was piloted in a team of 25 employees of a portal provider. An immense number of database processes are integrated and deleted offline. Monetary value generated, is simply to understand here, as a saving through the workload of the pilot employees. A great many files or documents were opened to trace a combination for certain class integration or to add as default repositories. The results indicated that the traditional linear theory with product data management (Bill of Material)-oriented implementation of product structuring becomes difficult and expensive. We stress here that the requirement has to be captured before product versioning in its physical mode or *invisible* format. The experimental outcome is given in Table 19.4:

Experimental results of a telecom system				
Item	Methodology before use	Methodology after use	% Time saved	US\$/1,000 K value generated
Semi-automated provisioning	Manpower employed mainly	Electronic format-model-based	40 – Regular man-hour	500
SCADA-Integration	Databased in different sources	Easy integration, no filtering, but models, interpretation and visualization	50 – man-hour	650
Customer requirements	Manually collected, data not used and storage capacity large	Totally modeled, integrated, agent-based browser access to view and integration to product offering	80 – man-hour	750

Table 19.4. *Experimental results of the pilot study at an IT-corporation*

On the other hand, a very small cause that escapes our notice determines a considerable effect that we cannot fail to see, and then we say that the effect is due to chance. The evaluation tool and tracking elements made life easier when implementing the project.

The inspiration for this project came from the initiatives of an IT-corporation in Finland, which failed in applying giant software which

were expensive and difficult for piloting purposes. What we learned here is that the most complex things in the known universe are living creatures, such as human beings. These complex systems are made from the most common raw materials known to exist. Those raw materials naturally assemble themselves into self-organizing systems, where simple underlying causes can produce surface complexity.

During the actual implementation of the product development project, working in parallel is essential. This requires flexible service-mindedness from all participants (marketing, R&D, production, logistics, customer service, etc.) in spite of the fact that the goal must be pursued through several iterations, on the basis of information that, at least in the beginning, may be somewhat inaccurate. The speeding up of a product development project is always so dependent on the product and type of task that its efficient implementation cannot be guaranteed without detailed preparatory work and constant replanning according to the current situation. Especially, in designing a modular product concept, we should aim at making the interfaces between the modules insensitive to changes in the adjacent modules. This can be achieved by choosing interface elements with weak interdependences related to those structural elements of the modules themselves that take the longest time to design.

19.15. Conclusions

Separating complicatedness and complexity improves the clarity by which systems can be described and analyzed. In this way, it is possible to clearly separate an inherent property of the system, complexity, from a derived attribute, which is complicatedness. The mathematical expressions were formulated to capture additional properties of systems that have heretofore remained largely unaddressed. It is possible to derive results that give valuable insight into the behavior of systems. These insights are useful in the analysis and design of very large complex systems and also move us towards a theory of complicatedness.

The speeding up of a product development project is always so dependent on the product and type of task that its efficient implementation cannot be guaranteed without detailed preparatory work and constant replanning according to the current situation. Especially in designing a modular product concept, we should aim at making the interfaces between the modules insensitive to changes in the adjacent modules. This can be achieved by choosing interface elements with weak interdependence related to those

structural elements of the modules themselves that take the longest time to design.

Industrial experimentations and cases shown at the end of the chapter illustrate and validate some of the mechatronics theories as science introduced in the beginning of the chapter.

19.16. Bibliography

- [AND 83] ANDREASEN M.M., “Integrated product development – a new framework for methodological design”, *International Conference on Engineering Design, ICED 83*, Copenhagen, Denmark, pp. 289–297, 1983.
- [CAR 93] CARRUBA F.P., “Designing people-pleasing products”, *International Conference on Engineering Design (ICED 93)*, The Hague, Netherlands, 1993.
- [COO 90] COOPER C.G., “New products – what distinguishes the winner”, *Research and Technology Management*, vol.33, pp. 27–31, 1990.
- [EPP 91] EPPINGER S.D., “Model-based approaches to managing concurrent engineering”, *International Conference on Engineering Design (ICED 91)*, Zurich, Switzerland, 1991.
- [FER 93] FERDINAND A.E., *Systems, Software and Quality Engineering*, Van Nostrand Reinhold, New York, 1993.
- [GHA 99] GHARAJEDAGHI J., *System Thinking: Managing Chaos and Complexity*, Butterworth-Heinemann Publishing, 1999.
- [HAB 06] HABIB M.K., “Mechatronics engineering the evolution, the needs and the challenges”, *Proceedings of the 32nd Annual Conference of IEEE Industrial Electronics Society (IECON 2006)*, pp. 4510–4515, 2006.
- [HAB 07] HABIB M.K., “Mechatronics: a unifying interdisciplinary and intelligent engineering paradigm”, *IEEE Industrial Electronics Magazine*, vol. 1, no. 2, pp. 12–24, 2007.
- [HAB 08] HABIB M.K., “Interdisciplinary mechatronics: problem solving, creative thinking and concurrent design synergy”, *International Journal of Mechatronics and Manufacturing Systems (IJMMS)*, vol.1, no.1, pp. 264–269, 2008.
- [HOL 90] HOLLINS B., PUGH B., *Successful Product Design*, Butterworths, London, UK, pp. 1–167, 1990.
- [HUB 88] HUBKA W., EDER W.E., *Theory of Technical Systems*, Springer-Verlag, Berlin, Germany, p. 275, 1988.
- [KOH 89] KOHONEN T., *Self-Organizing Map*, 3rd ed., Springer-Verlag, Berlin, Heidelberg/New York, NY, 1989.

- [LIN 01] LIN Y., ZHANG W.J., WATSON L.G., "On proposal of function-behavior-State framework as refinement of EID (ecological interface design)", in ARAI E. (ed.), *Human Friendly Mechatronics*, Elsevier Publication, Tokyo, Japan, pp. 61–71, 2001.
- [PIL] PILLAI B., "Adaptation of neural network and fuzzy logic for the wet-end-control and process management of paper or board achiness: a tool making approach", *Actia Polytechnica Scandinavica Publication*, Mechanical Engineering Series No. 138, Finland.
- [PRI 98] PRICKETT S.E., MAVROVOUNIOTIS M.L., "Construction of complex reaction systems, part 1–3", *Computational Chemical Engineering*, p. 15, 1998.
- [REI 91] REINERTSEN D.G., SMITH P.G., "The strategist's role in shortening product development", *Journal of Business Strategy*, vol. 4, pp. 18–22, 1991.
- [SAL 94] SALMINEN V., VERHO A., "Systematic shortening of the mechatronic product development cycle", *The International Conference on Machine Automation, Mechatronics Spells Profitability*, Tampere, Finland, 15–18 February 1994.
- [SAL 03] SALMINEN V., PILLAI B., "Methodology on product lifecycle challenge management for virtual enterprises", *4th IFIP Working Conference on Virtual Enterprises, PRO-VE'03*, Lugano, Switzerland, 2003.
- [SAL 00] SALMINEN V., PILLAI B., "Non-linear dynamics of interacting mechanisms in distributed product development", *International Conference on Complex Systems, ICCS 2000*, Nashua, NH, 2000.
- [SMI 91] SMITH P.G., REINERTSEN D.G., *Developing Products Half the Time*, Van Nostrand, New York, p. 287, 1991.
- [TAN 00] TANG V., SALMINEN V., PILLAI B., "Extending DSM to very complex systems. Design structure matrix", *World-Wide Conference*, Massachusetts Institute of Technology, Cambridge, Boston, MA, 2000.
- [TAN 01] TANG V., SALMINEN V., "Towards a theory of complicatedness: framework for complex systems analysis and design", *13th International Conference on Engineering Design, ICED 2001*, Glasgow, Scotland, UK, 21–23 August 2001.
- [TWI 80] TWISS B.C., *Managing Technological Innovation*, Longman, London, UK, p. 234, 1980.

Chapter 20

A Mechatronic Platform for Robotic Educational Activities

Robotics courses can be efficiently taught only if self-involvement and development are encouraged early on. Robotics in education is mainly focused on the different disciplines that are put into practice in the respective field, including programming, hardware development, mechatronics, etc. This chapter discusses an end-to-end procedure to develop an autonomous robotic platform, as part of the educational procedure. First, the architectural design and the implementation of the platform are described in detail and then the educational activities that can be applied to this platform are detailed. The goal of this chapter is to highlight methods to stimulate the interest of students in robotics through their involvement in the development of a robotic platform and, later on, of the required software applications. In a nutshell, this chapter summarizes our experiments in mechatronics educational activities that actively involved students.

20.1. Introduction

During the past few decades, persistent research endeavors in the areas of robotics and artificial intelligence allowed new educational activities to take place in a real environment. In a course on intelligent tutoring systems (ITSs),

Chapter written by Ioannis KOSTAVELIS, Evangelos BOUKAS, Lazaros NALPANTIDIS and Antonios GASTERATOS.

first introduced in [CAR 70], and considering the advanced robotic technology, the simulations and the human–robot interaction have been extended to the environment of the robots. By exploiting various sensors and actuators, robots are capable of perceiving and interacting with the real world. Taking advantage of this intrinsic capacity, the robotic technology puts into effect a series of educational activities that have been developed to aid and foster learning of the relevant topics. Consequently, tutors have been able to increase the motivation and performance of their students by means of diverse robot building activities, such as robotic competitions and algorithm testing [MUR 01, NOU 04].

The majority of the research conducted in robotic educational activities can be categorized according to the interaction they provide with the teaching procedure. This also includes the approaches that aim to teach subjects closely related to the robotics. For example, the work presented in [PET 04] provides an effective understanding of robot programming and engineering principles, while in [KLA 03] the use of Lego Mindstorms robots for lab exercises and projects in classes is proposed. In a different approach [JAN 04], the authors presented the use of robotic technologies, for example application of a robot to escape from a maze to teach Personal Digital Assistant (PDAs) programming. The reader may refer to [WAN 01, WEI 01, AVA 02, JAN 04, AHL 02] for activities that include similar paradigms in robot programming, robot construction, artificial intelligence, algorithm development, embedded systems and mechatronics. Additionally, the growth of interest in mechatronics has identified a need for the provision of engineers whose education and training enables them to operate in an interdisciplinary manner [HAB 06]. In all the aforementioned works, the robot is used in the educational activity as just a passive tool, which can be considered as either the result of an assembling process of the students or a physical machine that executes a set of instructions and algorithms also provided by the students. Yet, alternative viewpoints exist, where autonomous robots could be used as active mediators of the educational activity, where they have an active role in the learning process [MIT 08].

In all such educational activities there is a plethora of robots that can be utilized, either as parts that should be assembled or as an out of the box solution ready for programming. The LEGO NXT robotic kit has been used for the construction of robots with up to three degrees of freedom [BAG 07]. Concerning the robots that are ready for use, the most representative example is the NAO humanoid robot, where the mechanics and electronics are already assembled and ready to run. It has been widely used in RoboCup projects by various universities, where the goal is to develop a team of three robots playing soccer against other teams [MÜL 11]. A humanoid robotic platform with less capabilities is the Robonova-1, which brings together both the

construction of the platform and its programming to accomplish simple robotic applications [LUP 10]. Apart from the humanoid robots there are also mobile platforms that are available for programming and development of sophisticated navigation algorithms. We could start with the Boe-Bot robot platform, which requires only basic skills in robotics and is suitable for accomplishing simple activities [BAL 08]. A more advanced and complicated robotic platform that enables the training of the users to enhanced robot navigation capabilities is the Pioneer platform. By using such a robot, the authors in [ANG 04] were able to detect and model doors in an indoor environment. Similarly, the authors in [SMI 09] describe a mobile robot platform equipped with vision sensors that is suitable for data acquisition under the limitations of the manual guidance. Additionally, there are also platforms that combine attributes from both humanoid and mobile platforms such as the PR2, which has been used to carry out advanced robotic applications such as grasping, simultaneous localization and mapping (SLAM) and path planning [MEE 10].

The main goal of this chapter is to show how educational activity can benefit by the development of a generic mobile robot platform. The contribution of this work can be divided into two objectives. The first objective involves the description of the activities that took place during the development and construction of a mobile robotic platform, suitable both for exploration of indoor and outdoor environments. This part includes all the prerequisite steps, namely requirement specification, the AutoCAD models, the construction of the platform and the mounting of the utilized sensors. The second objective concerns the description of the educational activities that will occur by utilizing this robotic platform. In particular, it will be used as a testbed in a series of undergraduate and graduate courses to assess students' capabilities in mechatronics and vision as well as to assist the students in acquiring knowledge in the respective topics. More specifically, reference is made in the different robotic applications that are supported by this platform, such as three-dimensional (3D) reconstruction, visual odometry (VO), visual SLAM (VSLAM), human-robot interaction tasks as well as robot control routines and hardware implementation. The latter provides a good insight for the potential of such a platform and the variety of the robotic educational activities that are supported.

20.2. System overview

The proposed work targets the development of a robotic agent equipped with high-resolution vision sensors ensuring great navigation and perception capacities of its environment. To effectively design and implement such a

robotic platform, several specification requirements should be posed in the first instance. The system should provide the means to the students to effectively use specific robotic activities. More specifically, the robot should be equipped with a wide stereoscopic camera mounted on the top of the robot, suitable both for the perception of the environment and the 3D reconstruction of the surroundings. Additionally, it should be equipped with a narrow stereoscopic camera mounted close to the base of the robot aiming at the ground exactly in front of it, which will be suitable for navigation activities such as VO. This second stereo camera is indeed useful due to the fact that localization algorithms tend to produce more accurate location estimations in static environments, that is when the sole moving object in the observed scene is the robot [NAL 11]. Consequently, the closer the stereo camera to the ground the less the observation of non-static objects in the scene. Moreover, the robot should be also equipped with an active vision sensor, that is, Kinect, that provides reliable in-depth information of the scene and a robust sensory input for indoor navigation and human–robot interaction procedures.

20.2.1. Architectural design

The first step towards the design of a robotic platform is the creation of a draft, which will comply with the entire system setup, as the one illustrated in Figure 20.1a. The proposed architecture comprises a robotic platform, a Bumblebee2 stereo camera (narrow stereo), a custom-made wide stereo camera which is mounted on a mast, a pan-tilt unit that holds the wide stereo camera and a Microsoft Kinect sensor. The assembly study behind the draft that is shown in Figure 20.1a is the following:

- The wide stereo pair, built from individual monocular cameras, placed 1 m above the ground with fixed tilt angle of 39° .
- The localization stereo pair, which is based on the Bumblebee2 camera, placed 0.3 m above the ground with a fixed tilt angle of 31.55° .
- The Microsoft Kinect sensor mounted under the pan-tilt unit parallel to the robotic base.
- The entire vision system of the robot mounted on a specially designed metallic structure, mounted and stabilized over the base of the robot.
- The pan-tilt unit that allows the rotation of the wide stereo pair used to augment the vision capabilities of the wide stereo.

The next step comprises the drawing of the custom parts of the robot in a 3D Computer-Aided Design (CAD) environment. This task demands

mechanical skills and provides the opportunity for the students to design in AutoCAD and implement an integrated structure on a computer numerical control (CNC) machine tool. The virtual model of the designed robot is presented in Figure 20.1b.

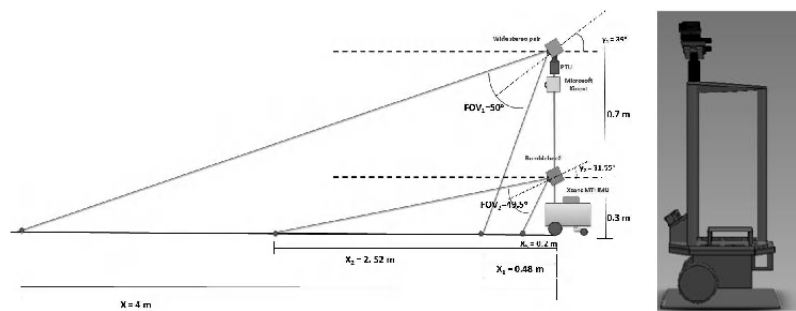


Figure 20.1. The robotic platform illustrated as a) schematic of an architectural design and b) conceptual draft with AutoCAD



Figure 20.2. a) The Videre Erratic mobile platform and b) the respective specifications

20.2.2. Subsystem equipment

20.2.2.1. Robotic base

An off-the-shelf robotic platform was chosen as the base of the robot to be constructed. The Videre Erratic mobile platform was used, which is a full-featured, industrial mobile robot base. It is a compact and powerful platform, capable of carrying a full load of robotic equipment, including an integrated PC, laser rangefinder and stereo cameras. The model of the platform is illustrated in Figure 20.2a, while the respective specifications

are analytically presented in Figure 20.2b. Moreover, the platform bears an integrated mini-PC that is capable of performing even computationally intensive tasks.

20.2.2.2. Robot skeleton

The skeleton of the robot is a custom-designed construction to bear all the sensors and equipment to be utilized. The first part of the skeleton is a metallic base that holds the Bumblebee2 camera in a specific position and orientation. More analytically, the base should hold the stereo camera 0.30 m above the ground ensuring a tilt angle of 31.55° . Additionally, the camera should be aligned with the front part of the robot retaining the consistency between the environment perception and the directions to the servo-motors of the platform, therefore, the construction of the metallic parts should permit accuracy better than 1 mm. A mechanical interface has been designed consisting of three different parts, two of which are asymmetric, outlined to result in the desired tilt angle. The proposed topology is illustrated in Figure 20.3a. The rest of the skeleton consists of three identical metallic poles, which are fastened by a frame screwed on the platform and a triangular plate on top of the poles. The latter is also utilized as the base for the wide stereo rig. It should be mentioned that the design and the implementation of the metallic construction has taken into consideration the maximum payload of the Erratic platform, which should be less than 20 kg. Therefore, the materials that has been utilized for the implementation of the skeleton is aluminum and the overall weight was approximately 5 kg. The metallic construction with the respective dimensions is shown in Figure 20.3b.

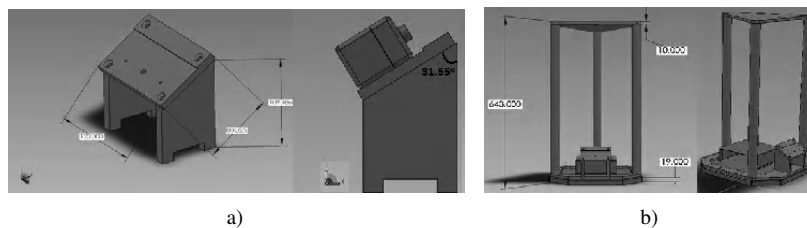


Figure 20.3. a) The base for the Bumblebee2 stereo camera and b) the entire metallic construction as designed in AutoCAD

20.2.2.3. Wide stereo camera

The wide stereo camera consists of two identical Prosilica GE1660C cameras, an example of which is illustrated along with its specifications in Figure 20.4. The construction of the stereo rig is a demanding procedure

as it involves the assembly of several different parts and also demands the specification of many parameters. The most important one is the regulation of the baseline distance between the two cameras. This is an application-oriented parameter and influences the range within which the resolution is better than or equal to a desired value, that is by increasing the baseline the range is also increased, given that the rest of the intrinsic parameters are constant [KOS 11]. Therefore, a worm screw was used to ensure the regulation of the baseline distance within the range [15, 22] cm. An additional difficulty in the construction of the stereo rig is the precise alignment between the cameras, that is to parallelize the two respective principal axes. Therefore, a pair of additional regulating devices (tip-tilt units) have been placed beneath each of the cameras that adjust the rotation angle between the two axes (roll and pitch). The divergence of the two cameras on the yaw rotational angle is not of great importance as it can be easily minimized by the subsequent stereo calibration procedure. The tip-tilt unit bears a precision micrometer drive and is mechanically compatible to other interfaces. We considered the application scenario where the wide stereo vision system needed to cover a view range between 0.48 and 4 m and a horizontal field of view (FoV) of 120° . The wide stereo vision system was placed on the pan-tilt unit that is fixed on top of the metallic skeleton construction. The pan-tilt device used is the Directed Perception PTU-D46. This device has a pan range of 318° ($-159^\circ, +159^\circ$) and a tilt range of 78° ($-47^\circ, 31^\circ$). Given the fact that the vertical FoV of the used cameras in the wide stereo setup is 50° , the stereo rig should be tilted by an angle of 39° to accommodate the desired view range. Since the cameras horizontal FoV is smaller than the specified 120° , multiple pictures have to be captured by panning the wide stereo camera. More specifically, a solution for a camera with such a FoV, three pictures should be taken with panning angles of -35° , 0° and 35° , respectively. This results in an overlap of at least 25% between each successive couple, which is sufficient for a correct alignment and merging of the images into a wider image covering the total 120° . The wide stereo accompanied with all the aforementioned subordinate devices is illustrated in Figure 20.4.



Figure 20.4. a) The Prosilica camera and b) the specifications of the Prosilica camera

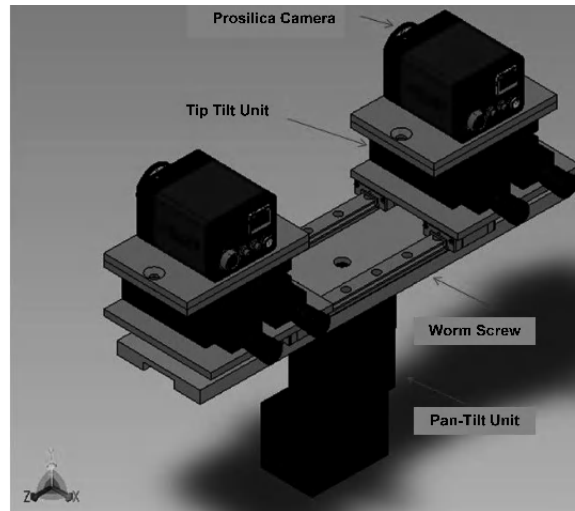


Figure 20.5. The AutoCAD model of the wide stereo head with the pan-tilt unit

20.2.2.4. Narrow stereo camera

For the narrow stereo setup, the Bumblebee2 camera has been utilized. This camera has been exploited in various robotic applications both for indoor and outdoor exploration [NAL 09]. It has a very compact construction and retains precise alignment between the two vision sensors. This stereoscopic camera has a baseline distance among the two cameras that is 0.12 m and, therefore, we call it “narrow stereo” for the proposed robotic platform. The camera along with its specifications is illustrated in Figure 20.6.

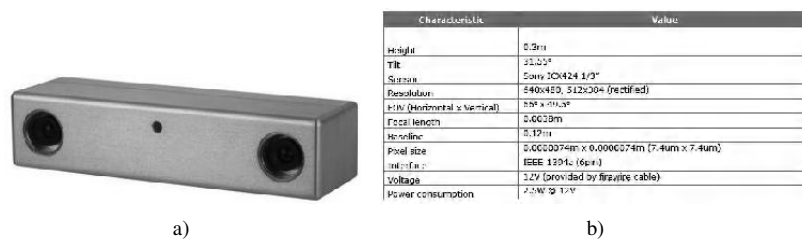


Figure 20.6. a) The Bumblebee2 camera and b) the specifications of the Bumblebee2 camera

20.2.2.5. Microsoft Kinect sensor

Active vision sensors, such as time-of-flight (ToF) cameras, laser scanners and the Microsoft Kinect, are widely used in vision-based robotic applications due to the fact that they are able to directly obtain the depth map of the scene, without the need to spend additional computational resources that a stereo vision system would demand [STU 11]. The proposed robotic platform is thus additionally equipped with a Microsoft Kinect sensor, which is mainly suitable for indoor applications, due to the fact that its results are strongly affected by daylight. It can be used as a motion sensor that provides a natural user interface allowing users to interact intuitively and without any intermediary device, such as a controller. Among other attributes, the Kinect system can identify individual users through face recognition and voice recognition. In general, the Kinect sensor is widely used in education, enabling students to develop human–robot interfaces and various robotic applications [TÖL 10]. This sensor along with the respective specifications is shown in Figure 20.7.

Parameter	Value
Field of View (Horizontal, vertical, Diagonal)	58° H, 46° V, 78° D Vertical : 43°
Depth image size	VGA (640 x 480)
Spatial x/y resolution (@ 2m distance from sensor)	3mm
Depth z resolution (@ 2m distance from sensor)	1cm
Operational range	0.8m ... 3.5m
Color image size	UXGA (1600 x 1200)
Data interface	USB 2.0
Power supply	USB 2.0
Power consumption	2.25 W
Dimensions (width x height x depth)	14cm x 3.5cm x 5cm
Operational environment	Indoor

Figure 20.7. a) The Kinect sensor and b) the specifications of the Kinect sensor

20.2.2.6. FPGA platform

The designed robotic platform also bears a field-programmable gate array (FPGA) device that is suitable for hardware programming and for parallelizing computational demanding tasks. An FPGA is a semiconductor device that can be programmed directly by students given the suitable software tools. As such FPGAs combine the best features of both hardware (i.e. speed and parallelization) and software (i.e. programmability and ease of update). Instead of being restricted to any predetermined hardware function, an FPGA allows the students to program their own features and functions, adapt to new standards and reconfigure the hardware for specific applications. FPGAs can be used to implement any logical function that an application-specific

integrated circuit (ASIC) could perform, but the ability to update the functionality after shipping offers advantages for many robotic applications that need to be performed in real-time. Therefore, the selected FPGA device is the Virtex-6 LXT, which comprises the third-generation advanced silicon modular block (ASMBL) column-based architecture shown in Figure 20.8.

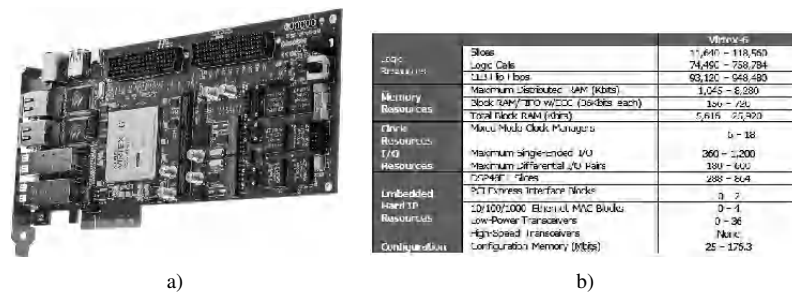


Figure 20.8. a) The Virtex-6 FPGA and b) the specifications of the Virtex-6 FPGA

20.2.2.7. Inertial measurement unit

On top of the metallic plate, behind the wide stereo head, a compact inertial sensor platform has been placed, which is able to measure translational and rotational movements about all three axes. The device used here is the MTi-G, a small size and low-weight Global Positioning System (GPS) aided MEMS-based inertial measurement unit (IMU) and static pressure sensor Figure 20.9. The platform possesses an internal signal processor to provide accurate drift-free signals with highly dynamic response and long-term stability. Moreover, the embedded GPS sensors renders this device an excellent navigation and control aid for outdoors environments. MTi-G smart design allows its utilization in a wide range of output modes and sophisticated settings for specific usage situations, optimizing the sensor fusion (s.f.) algorithm routine for different applications. Indeed, the use of the IMU on our test robotic platform is twofold: (1) to provide a heterogeneous to the visual source of localization measurement, thus to enable the application and testing by the student of robust s.f. algorithms and (2) to allow the implementation and testing of image stabilization algorithms by fusing the rate gyro signals from the roll and pitch angle of the inertial sensors platform and the optic flow vectors from the image planes of the two cameras.

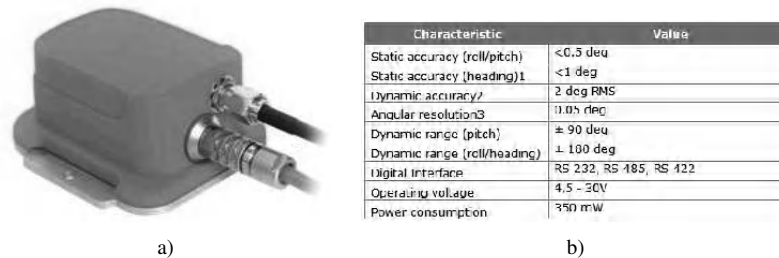


Figure 20.9. a) The MTi-G and b) the specifications of the MTi-G

20.2.3. Robot montage

This section describes the integration of all the aforementioned sensors and hardware devices into a single architecture, as shown in Figure 20.1a. This is a time-consuming task as it involves the efficient cooperation between software and hardware. As far as the hardware part is concerned, the entire construction should make use of mechanical interfaces designed to match exactly the sensors mountings. The entire construction has been assembled bearing an accuracy as high as possible. Specifically, the metallic parts have been designed and implemented with accuracy better than 1 mm. The wide stereo rig is a custom-made device and its accurate alignment is mandatory to produce accurate 3D reconstructions. Therefore, it has been implemented with an accuracy better than 0.1 mm. Moreover, the selected mechanical units (worm screw and tip-tilt) also retain strict specifications from the constructor. The exact knowledge of the precise positions of the sensors onboard the robot allows us to perform the accurate transformations that are demanded during the development of software applications, thus avoiding systematic errors that might occur due to misalignments. Concerning the software part, which is also important for the integration of the robot, several procedures have to be calibrated. The most demanding procedure is the calibration of the wide stereo camera setup, as it requires simultaneously the regulation of the roll and pitch rotation angles of the cameras. The epipolar error after the stereo calibration procedure that has been performed is 0.35° for the wide stereo camera and 0.15° for the narrow stereo camera (Bumblebee2), which in both cases is within the acceptable ranges, that is less than 0.50° [SZE 10]. The Kinect sensor has also been calibrated; this procedure is slightly different as it comprises the registration and mapping of the pixels between the color and infrared camera [SMI 11]. The integrated robotic platform is shown in Figure 20.10.

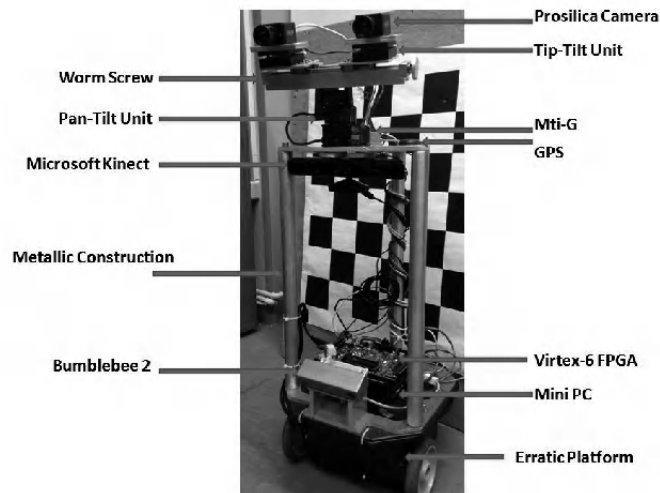


Figure 20.10. The integrated robot platform with its subordinate parts

20.3. Educational activities

The proposed robotic platform is suitable for a great variety of robot navigation activities, as it bears various sensors suitable either for indoor or outdoor exploration. More precisely, the platform is utilized for educational purposes on different courses, such as *Measurements* and *Robotics*. In accordance with these lessons, the students obtain hands on experience on data acquisition activities and robotic applications. This section describes the potential of the robotic platform to be used in various, but not limiting, application paradigms that are implemented and tested by the students.

20.3.1. 3D reconstruction

The 3D reconstruction can be performed either with the wide stereo camera if it concerns outdoor exploration or with Microsoft Kinect for indoor applications. In the outdoor exploration scenario, the stereo camera is accompanied by a stereo correspondence algorithm capable to produce dense disparity images, while in the indoor applications, where the Kinect sensor is utilized, the depth map of the scene expressed in meters is directly obtained by the sensor.

In case of the wide or narrow stereo configuration, the 3D reconstruction module is responsible for extracting the depth, or equivalently the

disparity, information from the stereo cameras arrangements and provide 3D representations of the scene. The image acquisition should be accompanied by a stereoscopic algorithm capable of computing consistent disparity maps [NAL 08]. Every stereo correspondence algorithm makes use of a matching cost function in order to establish correspondence between two pixels. The results of the matching cost computation comprises of the disparity space image (DSI). DSI is a 3D matrix containing the computed matching costs for every pixel and for all its potential disparity values. Usually, the matching costs are aggregated over support regions. These regions could be 2D or even 3D regions within the DSI cube. Due to the aggregation step it is not single pixels that will be matched, but image regions. Aggregation of the matching cost values is a common and essential technique in order to suppress the effect of noise that usually leads to false matching. The selection of the optimum disparity value for each pixel is performed afterwards. It can be a simple winner-takes-all process or a more sophisticated process. In many cases, it is an iterative process as discussed in section 20.3.1. An additional disparity refinement step is frequently adopted. It is usually intended to interpolate the calculated disparity values, giving sub-pixel accuracy or assign values to not calculated pixels. The general structure of the majority of stereo correspondence algorithms is shown in Figure 20.11. Moreover, a bidirectional consistency check is applied. The selected disparity values are approved only if they are consistent, irrespectively to which image is the reference and which one is the target. Thus, even more false matches are disregarded. If multiple complementary stereo pairs are available simultaneously, the disparity maps from each of them can be merged into a single larger depth map.

In the case of Microsoft Kinect, the disparity map is obtained directly from the sensor. More precisely, the Kinect sensor consists of an infrared laser emitter, an infrared camera and an color camera. The laser source emits a single beam that is split into multiple beams by a diffraction grating to create a constant pattern of speckles projected onto the scene. This pattern is captured by the infrared camera and is correlated against a reference pattern. The reference pattern is obtained by capturing a plane at a known distance from the sensor, and is stored in the memory of the sensor. When a speckle is projected on an object whose distance to the sensor is smaller or larger than that of the reference plane, the position of the speckle in the infrared image will be shifted in the direction of the baseline between the laser projector and the perspective center of the infrared camera. These shifts are measured for all speckles by a simple image correlation procedure, which yields a disparity image. For each pixel, the distance to the sensor is then retrieved from the corresponding disparity.

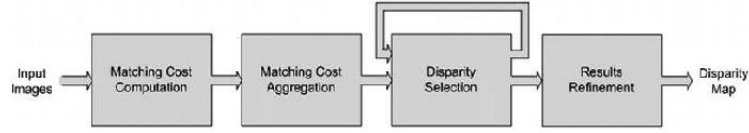


Figure 20.11. General structure of stereo correspondence algorithms

Once the depth map is computed in the case of the stereo rig or obtained immediately by the sensor in the case of Microsoft Kinect, then the reconstruction of the scene includes the transformation of the depth map to world coordinates by exploiting the triangulation. Specifically, by utilizing the depth information calculated in the disparity procedure, a dense point of the scene could be obtained and expressed in 3D world coordinates. More specifically, pixels expressed in camera coordinates $(x_c, y_c, disp(x_c, y_c))$, with respect to the stereo geometry, are transformed in 3D points (x, y, z) . The relation between the world coordinates of a point $P(x, y, z)$ and the coordinates on the image plane $(x_c, y_c, disp(x, y))$ is expressed in case we have a stereoscopic camera by the pin-hole model and the stereo setup as follows:

$$x = \frac{x_c \cdot z}{f} \quad [20.1]$$

$$y = \frac{y_c \cdot z}{f} \quad [20.2]$$

$$z = \frac{b \cdot f}{disp(x_c, y_c)} \quad [20.3]$$

where z is the depth value of a feature depicted in (x_c, y_c) , b is the stereo camera's baseline, f the focal length of the lenses expressed in pixels and $disp(x_c, y_c)$ the corresponding pixel's disparity value. In equations [20.1] and [20.2], x and y denote the abscissa and the ordinate in 3D world coordinates and corresponds to the (x_c, y_c) pixel in the image plane, respectively. The 3D reconstruction of a scene captured by a Microsoft Kinect is performed in a slightly different manner due to differentiations among the stereo and the RGB-D geometry. In the RGB-D scenario, a scale parameter is also considered in the triangulation procedure.

20.3.2. Visual odometry

The VO component estimates the robot displacement relative to a starting location. It is obtained by examining and correlating consecutive color images

captured by the narrow stereo or the Kinect sensor. The first step in a VO method comprises the detection of the salient landmarks between successive images. The success of the solely vision-based localization algorithms is, to a large extent, owed to the development of robust feature detection and description methods. The most common feature detector is the Harris corner detector while more recently developed feature detection methods such as the scale-invariant feature transform (SIFT), speed up robust features (SURF), and even the Center Surround Extremas (CenSurE) comprise both feature detection and description practices. A depth estimation of these features is then obtained by means of an enhanced stereo correspondence algorithm in case of the stereo rig or by exploiting the depth map obtained directly by the sensor in case of the Microsoft Kinect sensor. The next step comprises an outlier detection methodology responsible to discard both the mismatches between the features and the inserted errors due to the 3D reconstruction procedure. Assuming that the robot observes a specific point ${}^tP$ in 3D space, such as ${}^tP = [x_t, y_t, z_t]^T$. In time $t + 1$, the robot undergoes a specific motion with rotation ${}_{t+1}^tR$ and translation ${}_{t+1}^tT = [T_x, T_y, T_z]^T$, and the corresponding point ${}^tP$ is next observed as ${}^{t+1}P = [x_{t+1}, y_{t+1}, z_{t+1}]^T$. The transformation from point ${}^tP$ to ${}^{t+1}P$ is given as follows:

$${}^tP = {}_{t+1}^tR \cdot {}^{t+1}P + {}_{t+1}^tT \quad [20.4]$$

The calculation of the aforementioned transformation comprises the motion estimation procedure and maps the matched 3D reconstructed features between two successive time instances. In this module, only the correspondences that have passed the outlier detection procedure are utilized. Let us consider the resulting two 3D point clouds that correspond to times t and $t + 1$. The local coordinates feature position vectors ${}^{t+1}P$ on the reference image of the stereo pair (or the color image of the Kinect sensor) at time $t + 1$ are related to the position vectors ${}^tP$ in the reference image of the corresponding stereo pair (or the color image of the Kinect sensor) at time t according to the equation [20.4]. Ideally, six perfectly matched features should be sufficient to compute ${}_{t+1}^tT$ and ${}_{t+1}^tR$. However, in reality, error-suffering situations in several independent 3D points are needed for an efficacious calculation of ${}_{t+1}^tT$ and ${}_{t+1}^tR$, which should minimize a specific cost function, for example the least square one.

Students can obtain a deeper knowledge in the computer vision field with their involvement in the development of such a framework, as the VO can be divided into several subordinate modules, that is landmark detection, landmark 3D reconstruction, landmark matching and motion estimation. All the aforementioned routines constitute an active research topic in the field of robotics.

20.3.3. *Visual SLAM*

The visual SLAM problem is that of progressively building a map of the environment and at the same time localizing the robot on this map using vision. This module utilizes the output of the motion estimation and the calculated map to produce a global consistent map of the explored area. The robot could operate in indoor or outdoor environments, making different sensors more suitable for each scenario and posing different issues. The preferred sensor to obtain consistent and accurate depth maps could be, depending on the scenario, either one of the stereo camera setups or the Kinect sensor. The created map could be either a 2D grid in the simplest case or a 3D map in the most complex case. Additional sensors can be mounted on the robot by the students. This gives them opportunity to try out the use of different kinds of sensors to solve the SLAM problem. Other RGB-D sensors or ToF cameras or laser scanners can be used if the scenario favors them. In general, the different kinds of sensor provides heterogeneous outputs. Moreover, there are basic types of SLAM algorithms: the probabilistic and non-probabilistic SLAM algorithms. In our case, the students of undergraduate level, first use the non-probabilistic aspect as described in [NAL 11] obtaining a first experience of such a complex framework. Later on, on a more sophisticated course, which complies with the graduate level, the students use the probabilistic SLAM algorithms. As a result, students need to greatly understand the underlying principles of the system they work with before they can obtain meaningful representations of the world.

20.3.4. *Human tracking using RGB-D sensor*

This application comprises the detection and tracking of a human target from the robotic platform utilizing the mounted Microsoft Kinect sensor. The capabilities of this system concern, first, the recognition of the human body exploiting the skeleton detection and tracking application [BOU 11] and straight after the fixation on the trunk of the human body. Then, the mobile platform follows the human target by retaining a constant distance. The system decides about its movements by analyzing the data obtained directly from the Kinect sensor. The human is free to move in any direction and the robotic platform is responsible for retaining the human body at the center of the scene at a fixed minimum distance. The main limitation of the system is that in direct exposure to sunlight it influences the quality of the information grabbed from the infrared camera and the algorithm fails to execute correctly. In any case, the information derived from the RGB-D sensor is further processed on the embedded PC, which decides about the steering commands that are passed

to the wheels of the platform. Frequent controls for the distance between the platform and the human target ensure the direct response of the system in any movement of the human body. An example of this application is shown in Figure 20.12, where Figure 20.12a shows an instance from the color camera of the RGB-D sensor and Figure 20.12b presents the skeleton tracking processes, which is executed on the mini-PC of the mobile robot platform.



Figure 20.12. a) The color reference image of the Microsoft Kinect and b) the skeleton detection and tracking processes as illustrated on the mini-PC

20.3.5. Hardware acceleration

Autonomous robots behavior greatly depends on the accuracy of their decision-making algorithms. Vision-based solutions are becoming more and more attractive due to their decreasing cost, as well as their inherent coherence with human-imposed mechanisms. However, in the case of stereo vision-based navigation, the accuracy and the refresh rate of the computed disparity maps are the cornerstone of success. Robotic applications place strict requirements on the demanded speed and accuracy of vision depth computing algorithms. The proposed robotic platform is equipped with a Virtex-6 FPGA suitable for hardware implementation of the computer vision algorithms. In many applications [KOS 11], the hardware implementation of the vision algorithms is mandatory. This robotic platform provides the students with the opportunity to create even faster robotic applications by mapping the existing algorithms into VHSIC hardware description language (VHDL) and execute on an FPGA, which is easily integrated with the robot.

20.3.6. Sensor fusion algorithms

The data fusion exploits different data sources with regard to their synchronized use for the extraction of useful information. In the case, that the data sources derive from different sensors, then sensors fusion is concerned. This is a very useful tool in robotics enabling the utilization of diverging sensor data to extract accurate and robust measurements for navigation and control. Several expressions of the Kalman filter, fuzzy logic and Bayes law are among the s.f methods as shown in equation [20.5] that can be utilized and tested on a students project to assess their application in navigation and control. In equation [20.5], the $P(x, y, z)$ stands for the estimated robot position after the fusion of different sensors and estimation routines. Moreover, different sensor setups can also be assessed depending on the scenario. In particular, the navigation can be evaluated either with the narrow stereo camera, the GPS and the IMU for outdoor exploration or with Microsoft Kinect and the IMU for indoor applications.

$$P(x, y, z) = s.f.\{GPS(z, y, z), IMU(x, y, z), VO(x, y, z)\} \quad [20.5]$$

20.3.7. User interfaces

The development of the aforementioned algorithms by the students is performed in different interfaces. The algorithms are first created in high-level programming languages and right after are passed into robotic and application-oriented tools. We may start from the rapid prototype platforms such as the Matlab® to confirm the accuracy of the designed algorithms. Latter on more complex frameworks are designed by utilizing the LabView toolbox as well as generic compilers for hardware implementation. Once the developed algorithms have been accurately designed, they are integrated in the robot by utilizing the robot operating system (ROS), which is an interface that provides the capacity to the students to interact immediately with the robotic platform and the embodied sensors. The ROS provides libraries and tools to help in developing software applications for robots. It provides hardware abstraction, device drivers, libraries, visualizers, message passing and package management. ROS comprises a straight forward selection for robot integration applications as it retains an open source license. The fundamental concepts of the ROS implementation are nodes, messages, topics and services. A node is a process that performs computations and is meant to operate at a fine-grained scale; a robot control system will usually comprise many nodes. For example, one node can control the wide stereo camera, the Bumblebee2, the wheel motors, while some nodes are responsible for more

advanced robotic frameworks such as localization and path planning. Nodes communicate with each other by passing messages. A message is a strictly typed data structure. Standard primitive types (integer, floating point, boolean, etc.) are supported, as are arrays of primitive types and constants. Messages can be composed of other messages, and arrays of other messages, nested arbitrarily deep. A node sends a message by publishing it on a given topic, which is simply a string such as *odometry* or *map*. Topics have anonymous publish/subscribe semantics, which decouples the production of information from its consumption. Although the topic-based publish-subscribe model is a flexible communications paradigm, its *broadcast* routing scheme is not appropriate for synchronous transactions, which can simplify the design of some nodes. In ROS, this is called a service, defined by a string name and a pair of strictly typed messages: one for the request and one for the response. This is analogous to Web services, which are defined by uniform resource identifiers (URIs) and have request and response documents of well-defined types. Additionally, the ROS also provides a graphical view of the system, which can be utilized as a simulation tool in corresponding learning activities. The proposed robotic platform comprises computational nodes for each subordinate sensor, endowing the students with the capacity to create a complete application for robot navigation by combining the rent sensory inputs.

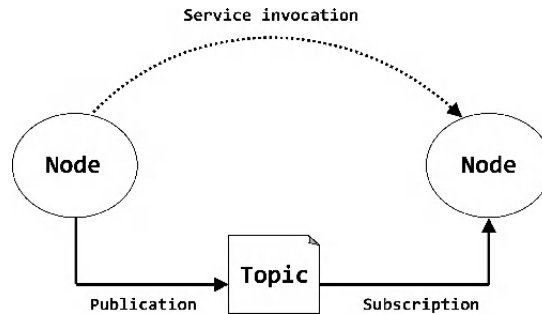


Figure 20.13. *The robotic operating system*

20.4. Experiences from educational activities

The developed robotic platform has been compared with similar constructions such as the one described in [SMI 09]. The specifications of the two compared platforms are summarized in Table 20.1, where the different sensor types are described. Specifically, the proposed system bears more sensors than the compared one, such as an extra stereoscopic camera, which

enables navigation with a different camera geometry setup, a Kinect sensor that enables the use of the platform in indoor environments and a hardware acceleration platform that provides the students with the capacity to involved with hardware development procedures. The presented platform lacks the existence of a panoramic camera that is a very useful tool to implement advanced image processing techniques such as panorama-stitching, however this issue could be bypassed with a top-down reprojection of the existing stereo pairs of the wide stereo rig.

The realized robotic platform has been utilized in various educational activities, which among other, includes the collection of data sets exploiting the vision sensors, which can be used for further assessment in other academic activities. So far three different data sets have been collected from the wide stereo camera, the Bumblebee2 and the Kinect sensor, respectively. Additionally, the robot setup has been evaluated by the students on already existing stereoscopic algorithms, such as the algorithm proposed in [NAL 09].

Sensor	Platform developed in Oxford New College	Proposed Platform
Robotic Platform	RMP200 from Segway	Videre Erratic mobile platform
Narrow Stereo	Point Grey BumbleBee	Point Grey BumbleBee
Wide Stereo	Non	Custom Made Stereo-Rig with 0.1mm accuracy
Panoramic Camera	LadyBug 2	Non
RGB-D Sensor	Non	Microsoft Kinect Sensor
Laser Scanner	Non	Non
Hardware Platform (FPGA)	Non	Virtex-6 LXT
IMU	Non	Mti-G
Pan Tilt Unit	Non	PTU-D46
GPS	CSI Series 5 Hz	Magellan ProMark-500 Rover
Operation	Outdoor	Indoor and Outdoor

Table 20.1. *The integrated robot platform with its subordinate parts*

Wide stereo camera. Several stereo pairs have been captured both in indoor and outdoor environments. The images acquired immediately from the cameras possess a resolution of 1,200 pixels, while after the calibration and rectification procedure the resolution was reduced to 1,120 pixels. Figure 20.15 depicts an indoor and outdoor scene captured by the wide stereo rig and the respective disparity images. Since the utilized stereo correspondence algorithm is a line-scan one, and given the fact that the quality of the resulting disparity images is very good, it can be deduced that the hardware regulation and the calibration procedure resulted in an accurately aligned stereo rig.



Figure 20.14. The first row depicts a stereo pair and the respective disparity of an indoor scene, while the second row depicts a stereo pair and the respective disparity map of an outdoor scene

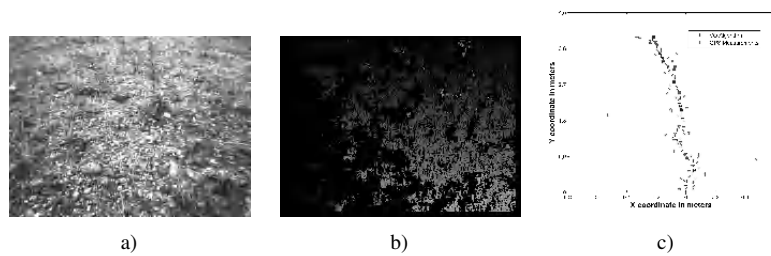


Figure 20.15. a) A left reference image of the Bumblebee2, b) the respective disparity map and c) the output of a VO algorithm along with the GPS measurements

Narrow stereo camera. The narrow stereo camera, that is Bumblebee2, has been utilized for the acquisition of a data set that corresponds to a short outdoor route. The main attribute of this data set is that it contains successive stereo pairs with close spatial proximity and, therefore, it is suitable for the evaluation of localization algorithms. It consists of 100 consequent stereo pairs and is accompanied by GPS location measurements. In the course of a robotic activity, this set of images has been employed by students for the evaluation of a 2D VO algorithm. Figure 20.15a and 20.15b depict a left reference image and a disparity map, respectively, while Figure 20.15c depicts the output of a VO algorithm along with the GPS measurements.

Kinect sensor. The Kinect sensor has been widely utilized as part of educational activities in indoor scenarios. A noteworthy task for students is the acquisition of a large data set known as the cognitive navigation data set, intended for use in indoor localization and semantic mapping. This data set has been captured with the Kinect RGB-D sensor mounted on the constructed robot, 0.8 m above the ground and tilted by 2.54° . It consists of three different parts:

- Figure 20.16a involves the representation of a corridor, an office, a laboratory, a bathroom and an elevator area, captured in natural illumination conditions. This means that the first part has been acquired in day light. This part contains 1,286 images including the respective depth maps.

- Figure 20.16b involves exactly the same places as Figure 20.16a, but it has been captured during night and, therefore, the illumination conditions are artificial. This part contains 1,292 images including the respective depth maps.

- Figure 20.16c has also been acquired under artificial illumination conditions and comprises a continuous exploration of all the aforementioned places. The movements of the robot are very smooth with common FoV between the successive frames. This data set contains a continuous sequence of 557 images and corresponds to a 70 m route.

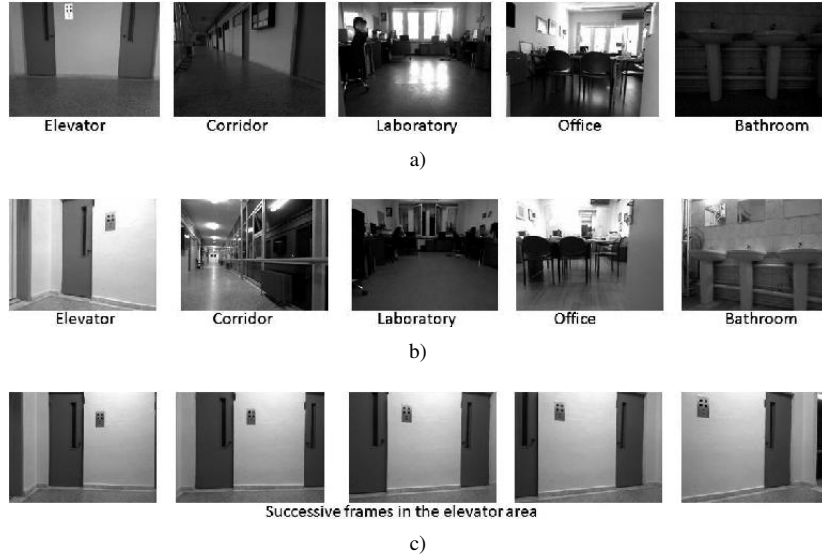


Figure 20.16. Image samples a) comprising natural illumination conditions, b) comprising artificial illumination conditions and c) comprising a small route of the robot in artificial illumination conditions

The first two parts of the data set are suitable for the validation of cognitive algorithms. The third part of the data set is suitable for indoor localization and mapping, due to the fact that the consecutive frames have close spatial and time proximity. Some examples of the captured images are shown in Figure 20.16. The cognitive navigation data set is freely available to the research community and can be retrieved, along with further technical information in [KOS 12].

20.5. Conclusions

In this chapter, the development of a robotic platform suitable for educational activities in indoor and outdoor scenarios has been presented. The work emphasized two directions. First, it goes through the description of the demanded steps that have taken place during the construction and assembly of the platform. This procedure has allowed students to be actively involved in the hardware development phases and includes the steps such as the requirement specification, the design of 3D CAD models, the construction of the platform and the physical placement and connection of the used sensors. Additionally, each utilized sensor has been described analytically by providing its specifications. The second focus direction concerns the description of the robotic educational activities that can be performed by exploiting the robotic platform. The integrated system will be used to assess students' capabilities in several topics of the mechatronics discipline. More specifically, reference is made in the variant robotic applications that can be supported from such a platform including 3D reconstruction, navigation, human–robot interaction tasks or even the low-level robot control routines. Moreover, some experiences of the educational activities have been shown providing the opportunity for the students to get involved and come one step closer to the state-of-the-art robotic technologies. Such student projects include data sets acquisition both in indoor and outdoor environments, while on the other hand, the students have exploited these acquired data to evaluate already existing algorithms, both for the perception of its environment and for robot navigation tasks.

20.6. Acknowledgments

The authors would like to thank the students of the eighth semester of Department of the Production and Management Engineering, Democritus University of Thrace, for their valuable help in the data set acquisition procedure. The authors would like to thank Mr. Nikos Rofalis for his active involvement in the project “Human tracking” using RGB-D sensor.

20.7. Bibliography

- [AHL 02] AHLGREN D., VERNER I., “An international view of robotics as an educational medium”, *International Conference on Engineering Education (ICEE'2002)*, Citeseer, 2002.
- [ANG 04] ANGUELOV D., KOLLER D., PARKER E., *et al.*, “Detecting and modeling doors with mobile robots”, *IEEE International Conference on Robotics and Automation*, IEEE, vol. 4, pp. 3777–3784, 2004.
- [AVA 02] AVANZATO R., ABINGTON P., “Mobile robot navigation contest for undergraduate design and K-12 outreach”, *ASEE Annual Conference Proceedings*, June 2012.
- [BAG 07] BAGNALL B., *Maximum Lego NXT: Building Robots with Java Brains*, Variant Press, 2007.
- [BAL 08] BALOGH R., “Basic activities with the Boe-Bot mobile robot”, *In Proceedings of Conference DidInfo*, 2008 FPV UMD Banska Bystrica, Slovakia 2008.
- [BOU 11] BOULOS M., BLANCHARD B., WALKER C., *et al.*, “Web GIS in practice X: a Microsoft Kinect natural user interface for Google Earth navigation”, *International Journal of Health Geographics*, vol. 10, no. 1, pp. 45, 2011.
- [CAR 70] CARBONELL J., “AI in CAI: an artificial-intelligence approach to computer-assisted instruction”, *IEEE Transactions on Man-Machine Systems*, vol. 11, no. 4, pp. 190–202, 1970.
- [CRA 04] CRAIG J., *Introduction to Robotics: Mechanics and Control*, Prentice Hall, 2004.
- [HAB 06] HABIB M., “Mechatronics engineering the evolution, the needs and the challenges”, *32nd Annual Conference on Industrial Electronics, IECON 2006*, IEEE, pp. 4510–4515, 2006.
- [JAN 04] JANSEN M., OELINGER M., HOEKSEMA K., *et al.*, “An interactive maze scenario with physical robots and other smart devices”, *2nd IEEE International Workshop on Wireless and Mobile Technologies in Education*, IEEE, pp. 83–90, 2004.
- [KLA 03] KLASSNER F., ANDERSON S., “Lego MindStorms: not just for K-12 anymore”, *IEEE Robotics & Automation Magazine*, vol. 10, no. 2, pp. 12–18, 2003.
- [KOS 11] KOSTAVELIS I., BOUKAS E., NALPANTIDIS L., *et al.*, “SPARTAN system: towards a low-cost and high-performance vision architecture for space exploratory rovers”, *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)* 2011, IEEE, pp. 1994–2001, 2011.
- [KOS 12] KOSTAVELIS I., GASTERATOS A., “Cognitive navigation dataset, group of robotics and cognitive systems”, Available at <http://robotics.pme.duth.gr/kostavelis/Dataset.html>, 2012.

- [LUP 10] LUPIÁN L., ROMAY A., MONROY P., *et al.*, “Cyberlords robocup 2010 humanoid kidsize team description paper”, *Workshop Robocup Singapore*, June 2010.
- [MEE 10] MEEUSSEN W., WISE M., GLASER S., *et al.*, “Autonomous door opening and plugging in with a personal robot”, *IEEE International Conference on Robotics and Automation*, IEEE, pp. 729–736, 2010.
- [MIT 08] MITNIK R., NUSSBAUM M., SOTO A., “An autonomous educational mobile robot mediator”, *Autonomous Robots*, vol. 25, no. 4, pp. 367–382, 2008.
- [MÜL 11] MÜLLER J., LAUE T., RÖFER T., “Kicking a ball-modeling complex dynamic motions for humanoid robots”, *RoboCup 2010: Robot Soccer World Cup XIV*, Springer, pp. 109–120, 2011.
- [MUR 01] MURPHY R., “Competing for a robotics education”, *IEEE Robotics & Automation Magazine*, vol. 8, no. 2, pp. 44–55, 2001.
- [NAL 08] NALPANTIDIS L., SIRAKOULIS G., GASTERATOS A., “Review of stereo vision algorithms: from software to hardware”, *International Journal of Optomechatronics*, vol. 2, no. 4, pp. 435–462, 2008.
- [NAL 09] NALPANTIDIS L., KOSTAVELIS I., GASTERATOS A., “Stereovision-based algorithm for obstacle avoidance”, *Intelligent Robotics and Applications*, Springer, pp. 195–204, 2009.
- [NAL 11] NALPANTIDIS L., SIRAKOULIS G., GASTERATOS A., “Non-probabilistic cellular automata-enhanced stereo vision simultaneous localization and mapping”, *Measurement Science and Technology*, vol. 22, no. 11, p. 114027, 2011.
- [NOU 04] NOURBAKHSH I., HAMNER E., CROWLEY K., *et al.*, “Formal measures of learning in a secondary school mobile robotics course”, *Proceedings of the IEEE International Conference on Robotics and Automation 2004 (ICRA’04)*, IEEE, vol. 2, pp. 1831–1836, 2004.
- [PET 04] PETRE M., PRICE B., “Using robotics to motivate ‘back door’ learning”, *Education and Information Technologies*, vol. 9, no. 2, pp. 147–158, 2004.
- [SMI 11] SMISEK J., JANCOSSEK M., PAJDLA T., “3D with Kinect”, *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, IEEE, pp. 1154–1160, 2011.
- [SMI 09] SMITH M., BALDWIN I., CHURCHILL W., *et al.*, “The new college vision and laser data set”, *The International Journal of Robotics Research*, vol. 28, no. 5, pp. 595–599, 2009.
- [STU 11] STURM J., MAGNENAT S., ENGELHARD N., *et al.*, “Towards a benchmark for RGB-D SLAM evaluation”, *Proceedings of the RGB-D Workshop on Advanced Reasoning with Depth Cameras at Robotics: Science and Systems*, California, USA, July 2011.
- [SZE 10] SZELISKI R., *Computer Vision: Algorithms and Applications*, Springer, 2010.

- [TÖL 10] TÖLGYESSY M., HUBINSKY P., “The Kinect sensor in robotics education”, *Proceedings of 2nd International Conference on Robotics in Education*, Austria, 2010.
- [WAN 01] JERRY B., WEINBERG GEORGE L., ENGEL, KEQIN GU, *et al*, *A Multidisciplinary Model for Using Robotics in Engineering Education*, American Society for Engineering Education, 2001.
- [WEI 01] WEINBERG J., ENGEL G., GU K., *et al*, “A multidisciplinary model for using robotics in engineering education”, *ASEE Annual Conference and Exposition*, Albuquerque, New Mexico, June 2001.

Chapter 21

The Importance of Practical Activities in the Formation of Mechatronic Engineers

The globalized job market needs engineers who have practical knowledge, and who can solve problems quickly, when they start their professional activities. Generally, it is very difficult to find newly qualified engineers who have the above characteristics. Traditionally, higher education focuses principally on theoretical training. Despite the difficulties encountered in changing the traditional structure of higher education, in which there has always been a clear dissociation between theory and practice, universities are now trying to improve the quality of teaching, and to meet the demand of the current job market. Practical activities are a differential that adds quality to newly graduated mechatronic engineers. It is essential that the students participate in extracurricular activities such as research projects, and visit companies involved in their specific field of activities. Thus, this chapter presents and discusses the importance of curricular and extracurricular practical activities for the training of mechatronic engineers, in order to prepare them for the current demand of the globalized job market.

21.1. Introduction

Since the beginning of civilization, humanity has dreamt of a better life. This has often been done through projects and the creation of systems to

simplify life, to seek pleasure and wellbeing and thus to promote what we call “progress” and “development”.

This development has been possible, thanks to new inventions, technological advances, scientific discoveries and advancements in all areas of human knowledge.

It can be affirmed that in modern human life, in any environment, most of the elements therein, have directly or indirectly been produced from the knowledge of engineering. In this sense, a great challenge to the formation of engineers is related not only to their technological knowledge, but also to their preparation in order to be able to contribute to the solution of urgent problems by applying technological innovation to improve the quality of life and to promote peace.

Among the engineering courses, mechatronics engineering, by its own definition, requires a horizontal training of professionals, as it involves various areas of knowledge. A scheme that very well characterizes all these involved areas of knowledge was proposed by Professor Kevin Craig as shown in Figure 21.1. This means that mechatronic engineers must have a strong basic formation in mechanical engineering, electro-electronics, control systems and computing. Knowledge of these systems can be used in the development of mechatronic systems for application in many areas of modern life. Mechatronic engineers also use knowledge of materials and of mechanical sciences associated with measurement, management and production techniques to create various types of products.

In discussing the teaching of engineering, and especially of mechatronics engineering, in a globalized world where unprecedented technological changes are causing transformations in our society, leads us to rethink pedagogical practice for teacher formation and the formation of engineering professionals. Technology education has recently been the target of vehement criticism and questioning. This may be due to the fact that we are using the educational models in a rapidly changing world, which have been created in the last century.

It would be reasonable then to assume that education technology should discuss, in parallel with specific contents, science, technology generation, its doubts, fears and the impact that it has on all of us. Unfortunately, this is not what is perceived by teachers, students, professionals and other representative sectors of our society.

The historical and social dimensions in the understanding of science and technology must be introduced. Despite the importance given to scientific and technological knowledge, much of the world's population still goes through unjustifiable problems and needs, even when the technical possibilities to solve them are available. We can imagine then that reflections and adjustments in the technological education process will contribute significantly to improving this situation.

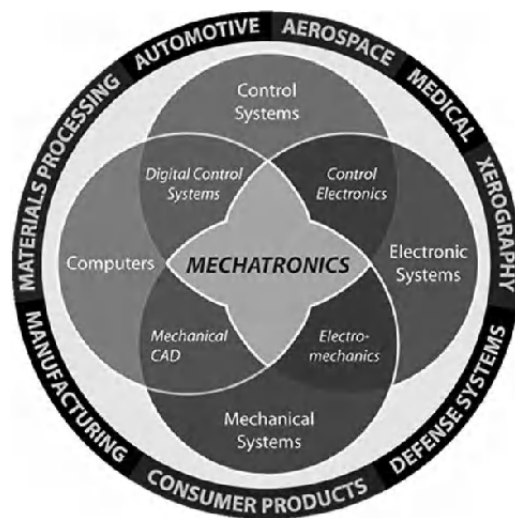


Figure 21.1. Definition of mechatronics (Professor Kevin C. Craig; Greenheck Chair in Engineering Design & Professor of Mechanical Engineering; College of Engineering, Marquette University, Milwaukee, Wisconsin 53201-1881; <http://multimechatronics.com>)

Engineering schools and universities have a great responsibility to organize their curricula so that their graduates and the population in general can take advantage of the speed of technical development. In general terms, the great challenge to the curricula is to prepare young engineers simultaneously in the fields of physics, mathematics, computer science, the environment, the humanities as well as in the specific area of formation in mechatronics engineering. Mechatronics engineering entails the integration of expertise in mechatronics engineering, electrical-electronics, computing and control systems. The curriculum should not be focused only on solving specific industrial problems, or solely on training the students to be mere users of existing technology [CAR 11, HAB 06, HAB 07].

The solution to this problem is not simple, especially considering that many products that we use today did not exist until recently. So, what the population will need in the near future should be included in the development of a mechatronics curriculum. Since the answer is not simple, Bevilacqua [BEV 09] summed up our situation as: “We are simultaneously observers and former users of objects displayed in a Science and Technology Museum”.

Another problem is putting a curriculum in practice: giving the students the most important information about the area in such a short time. Moreover, the curriculum must also take into consideration cultural, national and institutional conditions.

Howell [HOW 09] says that a curriculum in engineering may diverge from considering important aspects: “strong fundamentals vs. innovation and problem solving; individual excellence vs. team interactions; preparation for academic vs. industrial careers; research vs. application; solution of “traditional” vs. open-ended problems; and others. The best curriculum for one institution or country may be very different from another, depending on the perceived relative importance of these factors”.

Indifferently to the type of formation or curriculum, young mechatronic engineers must have a well-founded formation to be able to work in multidisciplinary areas, develop creative ideas and be ready to face new complex and interdisciplinary problems.

In this chapter, the authors discuss the issue of inclusion of practical activities in mechatronics engineering education. Some aspects correspond to professional experiences in developing the curriculum of mechatronics engineering at the Federal University of Uberlândia, Brazil. Teachers for approximately 30 years also have experience as course coordinators of engineering and in discussions in this area as members of ABCM – Brazilian Society of Engineering and Mechanical Sciences and IFToMM – The International Federation for the Promotion of Mechanism and Machine Science. Then, as the development of a curriculum in mechatronics engineering is not a “black box” where the contents are already predefined and defined, it is subject to controversy and criticism.

The systematization of technical education in Brazil has in its history models of French technical schools from the 17th and 18th Centuries: The *Académie Royale d'Architecture* (1671), the *École Nationale des Ponts et Chaussées* (1747) and *École des Mines* (1783). These schools represent the

first “civil” engineering schools in the world. In Brazil, technological education was introduced by the Portuguese in the 19th Century.

Until the 17th Century, it was the responsibility of the school to enable people to deal with subjects such as reading, writing, calculations, religious dogmas, philosophies and civil laws, according to Petitat [PET 94]. Technical education was introduced in the 17th Century. This teaching model was independent of traditional ways, which began the approach-applied work that consisted of an extension of technical and scientific practices in schools.

A great novelty introduced by the technical schools was education beyond things in themselves (objects and phenomena of nature), and much closer to theoretical models (mainly based on mathematics), that is idealized representation. Thus, a technical-scientific discourse was established allowing a practice of observation that penetrated experimentation in teaching. It is interesting to note that modern science took shape at the time of the Discourse on Method, by René Descartes, and the *Principia* by Isaac Newton.

In the first engineering schools, formation was more geared to specialized functions for the state, not for private productive systems. Thus, the state monopolized the new process of formation of technical professionals with power posture knowledge and a certain autonomy. It is in this context that these schools arise and establish themselves, causing and affecting changes in the educational system [CAR 03].

Engineering education in Brazil has foundations linked to the positivism of Auguste Comte. In the 19th Century, Brazilian engineers participated actively in the discussions between orthodox positivists (willing to promote a profound moral reform of society) and heterodox positivists (concerned with the definitive establishment of positive scientific knowledge in different areas). Most of these engineers opted for the second aspect, and it is thus inherited today and revered as a premise by individuals with technical formation. Heterodox positivists part from the understanding that students are empty containers having no knowledge; the teacher will fill them with their experiences and treatment of scientific knowledge, which is ultimately necessary for the intellectual pretensions of the human species [BAZ 00].

Engineering education is characterized by great fragmentation and hierarchy, especially in Brazil. Courses are divided into approximately two cycles, the basic and the professional, or when establishing sequences there

are very strict prerequisites among the various disciplines giving them a rigid sequence.

When it comes to engineering education, approaches and challenges related to the usual model of education reveal an amateurish posture that is often devoid of the rigor reserved for other professional procedures. Sporadic surveys conducted by individual educators worried about the problems of engineering education are, not rarely, devoid of theoretical analyzes that can deliver more consistent, realistic and promising new developments.

In engineering courses, the formation of technically capable individuals with a social, critical and creative vision is not suitably realized. Once this fact is verified, discussions among educators revolve around attempts to schedule a balanced distribution of technical content during the semesters. This task performed without proper diagnosis and without any didactic-pedagogic foundation (at least theoretical) will show, for certain, a certain distance between the desirable and the practical performance of everyday life. In general, the teachers are engineers, and the majority of them have postgraduation experience in engineering, but has no didactic formation.

Generally, the engineering curriculum is separated into two parts. The “basic cycle” that aims to “give” the students the fundamentals required for the next cycle. In some institutions, the “basic cycle” is offered separately from the institution responsible for the formation of engineers. It is not uncommon for the teachers of the basic cycle to lack the knowledge of how the topic given in their discipline will be applied in engineering. In practice, it has been observed that these contents have frequently been considered as if they were an end in themselves with no logical application in engineering. As in the professional cycle, in many situations, the informative process is given more importance than the formative process, and presupposes the knowledge given in the basic cycle and its projection for future professional performance; a projection that every teacher has regarding the market that is often stereotyped.

Despite attempts to improve and change the curricula of engineering courses, what is clear is that from the outset there is a blatant dissociation between theory and practice [BRI 93], focusing a strong theoretical content at the beginning of the course.

It may be noted that teachers in some countries have been concerned about this problem, focusing on efforts to improve their teaching programs in engineering. However, it is often possible to note that their concern is much

more about the quantitative aspect than the qualitative aspect and lacks curriculum consistency.

In this context, the importance of curricular and extracurricular practical activities in the formation of mechatronic students that should enable them to be more participatory and creative will be presented.

21.2. Curricular and extracurricular practical activities

In this chapter, practical curricular activities are obligatory under the flowchart of the course, and extracurricular activities are those in which students can participate, or not, depending on their affinity and/or interest.

The practical activities allow engineering students to learn, making them critical, creative agents interested in staying up to date to meet the challenges of their profession.

In general, some advantages of the practical activities can be summarized as follows: to consolidate formation in the basic disciplines of the course, either in specific activities or in interdisciplinary activities, thus enabling multidisciplinary integration; to promote the understanding of fundamental scientific principles and their role in the structure of engineering and to develop communication skills and relationships; to develop the habit of self-improvement and education after graduation; to develop the ability to create and to improve systems and methods; to develop the ability to work in groups to solve engineering problems that encompass technical, economical, political, social, ethical and environmental aspects.

Consequently, mechatronic engineers, so formed, may act in the design, implementation, operation and maintenance of automated production units and development of intelligent products taking into account the economics, management, security and environment. In addition to technical formation, university education should prepare professionals to be able to think and act by themselves. They must also have initiative and be prepared to take responsibility both at the social level and in their specific area of expertise.

It should also be noted that it is important for students to get experience in practical activities from the beginning of the course in order to encourage them to develop their knowledge in their area of expertise.

The following section presents some practical activities that contribute to the formation of young engineers.

21.2.1. *Practical laboratory activities*

In general, practical classes that are conducted in laboratories are associated with academic disciplines, even when they are separately included in the flowchart of the course.

The practical activities conducted in the laboratory aim to complement the content of lectures and learning. To achieve its goals, this activity should not only be in the demonstrable form. The students should be given the opportunity to effectively perform the planned activities. For good results, two students should work on each bench directly monitored by the teacher or the person responsible for the activity.

It is debatable whether it is better to have the classes conducted in research laboratories or in specific class laboratories. Each option has its advantages and disadvantages.

The research lab has the great advantage for students in letting them experience *in loco* new technologies and procedures. Being in the place where research is carried out, the students would have the opportunity to use ultramodern equipment. However, there has to be good programming because laboratory classes would interrupt research activities. Another disadvantage is that the students are not allowed to remain in the laboratory at the times that are not provided for their activities.

In turn, although the specific laboratories for teaching activities have a place and a time reserved for course activities, in general the components used by the students are not the same as those used in industry. This conveys a sense of disassociation from what is really used in practical applications in industry. Another feature that has been noted is that these labs typically have their installation and equipment underutilized. In this sense, it is important that the space should be designed to meet the various laboratory practices, and that the entire installation, including the support staff, should be well organized. Another aspect, although more difficult to implement, is that young engineers should have the opportunity to use industrial components that will be necessary for their professional lives.

21.2.2. Monitoring

Monitoring is an activity in which student-monitors help teachers to guide other students, to solve problems. To get good results in this teacher–student association, it is necessary that the student-monitor should have passed successfully the course in the particular discipline and know how to relate well with other students.

This activity allows the student-monitor to deepen his knowledge in the discipline, to show initiative, and to communicate and express himself with ease.

21.2.3. Research initiation activities

In research or scientific initiation activities, the student is enrolled in a research project coordinated by an experienced research professor. The students should be given assignments that must be developed in a given time. When involved in such assignments, the student has the opportunity to live with scientific procedures. This activity greatly develops the formation of the students, as they should always be studying, learning and incorporating new knowledge; the activity also helps them to be at ease in communication and expression, both in the written and spoken form; it should also permit them to have individual and teamwork leadership, and the ability to use computer technology to solve engineering problems in a systemic way.

This type of activity can be developed from the beginning of the course as many jobs consist, for example, of graphical and numerical simulations, where the student can easily learn a new programming language.

21.2.4. Participation in junior companies

The idea of the Junior Company emerged in France in the 1960s. In Brazil, the first junior company was founded in 1988.

The junior company is a non-profit organization formed and run exclusively by graduate students in schools of higher education. Its main objectives are to complement and to diversify the formation of the students to help them to practice classroom theory.

They serve mainly micro- and small businesses because they have a cost much lower than the market. To guarantee good work results, projects are monitored and supervised by an expert who has knowledge in the area of the project.

The junior companies have great advantages because they have the support of laboratories, technicians and teachers who work with high-quality technology, during the execution of the projects.

Students who have worked in junior companies have an advantage in entrepreneurial practices, management and business skills. Their contact with companies gives them experience in the form of expression and communication.

21.2.5. Academic mobility

Academic mobility gives the students a good chance to get new experiences and to acquire new knowledge, not only technically but also culturally. It gives them experience of new realities in their country and in foreign countries without hurting their academic careers.

We can cite two programs that give scholarships to enable students to conduct part of their training in other countries: “E4UM Erasmus Mundus Master in Mechatronic and Micro-Mechatronic Systems” is a two-year program that is conducted in three participant institutions. In the first year, the student selects an institution, does a basic discipline course, a language course and learns about the local culture. In the second year, the student moves to a second institution where he/she attends specialization courses. The three institutions are École National Supérieur de Mécanique et des Microtechniques ENSMM, France; Hochschule Karlsruhe – Technik und Wirtschaft HSKA, Germany and Universidad de Oviedo UNIOVI, Spain (<http://www.eu4m.eu>). The second program is the “Science without Borders” and it involves interchanging the study programs with many countries; it is sponsored by the Brazilian government. Thus, the program aims not only at the mobility of graduate students, but also at the mobility of researchers and postgraduate students, in addition to the participation of companies (<http://www.cienciasemfronteiras.gov.br>).

These programs are aimed at training highly qualified personnel, expanding innovative knowledge of industry staff and attracting new talent for research.

21.2.6. Participation in competition teams

The participation of students in competitions has been increasingly pervasive in universities, as it consists of the application of technical expertise to leisure activities. In general, teams have a teacher-tutor who coordinates and orients the work, but students are responsible for developing the competition system, projects details, construction/fabrication planning and tests as function of the competition program (with dates previously defined), elaborate technical reports, do oral presentations to defend their project, justify each adopted solution and maintain contact with potential sponsors for financial viability.

Participants in this activity deepen their technical formation and acquire other expertise that is not treated in the course, develop skills such as teamwork, leadership, capacity planning and project management, communication and expression ability, and the capacity to sell ideas and projects.

21.2.7. Participation in student directories

The student or academic directory is an association formed by students and aims to defend their interests, to promote rapprochement among students, teachers and faculty administration staff; to develop university spirit within and outside the school, to contribute to the prestige of their institution; to preserve the traditions and integrity of student academic life; to fight for the improvement of democratic institutions, to promote meetings of civic, social, cultural and scientific character, aimed at complementing and improving university education. Additionally, the academic directory aims to support students by involvement in social and technical projects, so that they can have good technical and social formation.

In general, the academic directory tries to give special attention to the new students and to promote greater integration between them and the senior students of the course.

It can be observed that the students involved in academic directories have a political vision much superior to that of the other students.

21.2.8. Participation and organization of events

From the work done by students in scientific initiation, in practical activities and in junior companies among others, students can organize events in their area. In such events, business representatives and researchers teach mini-courses that complement the training of students.

This type of organization, which may be local or national, allows them to acquire a great capacity for planning and management, and to develop their communication and expression skills.

In general, this type of organization is established by the initiative of the academic directories and the junior companies.

Participation in events organized by other institutions is a great opportunity which students have to divulge their work, to open space for reflection and discussion of the socio-political context, of their own profession as well as to increase their range of relationships.

21.3. Undergraduate course of Mechatronics Engineering at the Federal University of Uberlândia/Brazil

The undergraduate Mechatronics Engineering course at the Federal University of Uberlândia UFU, Brazil, is a semi-annual course, expected to be completed in 10 semesters having a total workload of 4,170 h. The course began in the first half of 2004 and is under the responsibility of the School of Mechanical Engineering FEMEC (www.mecanica.ufu.br). It should be noted that the School of Mechanical Engineering is responsible for the undergraduate courses in mechanical, mechatronics and aeronautical engineering, in addition to the postgraduate courses (master and doctorate degree) in mechanical engineering.

The participation of the students in practical activities, except in those of the laboratory classes that are obligatory for completion of the curriculum, has been encouraged by the promotion of educational and research scholarships that are obtained by the students through projects submitted to state agencies.

In order to obtain scholarships, teachers have to be dedicated to the course, because they play a fundamental role in applying for scholarships and in the coordination of the students' work.

The coordinator and the collegiate of the course have to do all the monitoring and execution of the study program during each semester and accompany the participation of the students in their various activities.

The following are comments on the implementation of practical activities in mechatronics engineering. The next section will give a general evaluation of the course that has been functioning for eight years.

21.3.1. *Practices activities of laboratory*

To prepare the political-pedagogical project of the mechatronics engineering course, we tried to obtain a significant number of laboratory practical classes, always associated with theoretical classes; this resulted in about 20% of the total workload of the course.

Because of the characteristics of the course and its teachers, some practice classes are taught in specific didactic laboratories and others in research laboratories.

In order to optimize staff support and the use of the physical space, the same location is used by more than one discipline. In this case, correct planning must be done in order to have a schedule that suits both the teachers and the students.

The specific didactic laboratories, in general, are related to the basic disciplines such as physics and electronics, and almost all of these use commercial components and have the support of the technical staff to organize the infrastructure for classes.

In the practical classes taught in research laboratories, the number of students allowed to participate depends on the space available and on the quantity of equipment to be used simultaneously. In these cases, the teacher plans his/her class and divides the students into groups so that they can participate in all the planned activities.

In most laboratory classes, there is a maximum of 12 students. This facilitates the teacher's work and allows the students to participate effectively in practical activities.

Figure 21.2 presents an experimental bench used in the practical classes of basic and power electronics. In this laboratory work, six students are

present in each group. Figure 21.3 shows the bench for the practical activities of hydraulic and pneumatic controls and industrial automation using industrial communication networks.

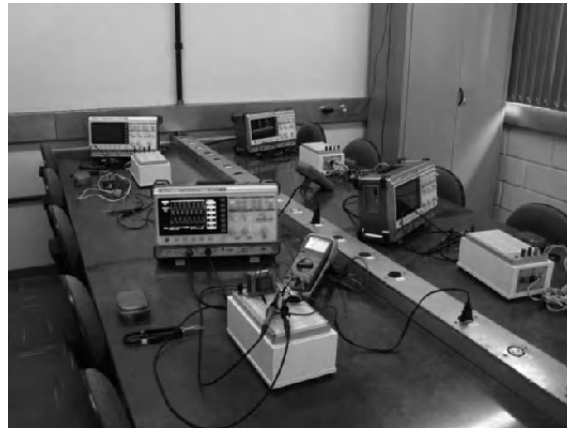


Figure 21.2. *Photo of didactic laboratory for activities on basic and power electronics*



Figure 21.3. *Photo of didactic laboratory for activities on hydraulic and pneumatic command systems, and industrial automation*

21.3.2. Monitoring

The Federal University of Uberlândia has a monitoring program with and without scholarships. The monitor contract is for six months and at its end, the student has to submit a report of his/her activities.

Mechatronics engineering students, in general, participate in the first periods of the monitoring program in which there is a strong focus on the disciplines of mathematics (differential and integral calculus, analytical geometry and linear algebra). This can be explained, at least partly, because in the later periods the students give preference to other activities related to their specific training.

21.3.3. Research initiation activities

Mechatronics engineering students (with or without scholarships) at the Federal University of Uberlândia have a strong participation in scientific initiation activities. Most scholarships are paid by development agencies and private companies.

Students participate in scientific initiation activities from the earliest periods of the course. Although the number of students varies, more than one-third are usually involved in it.

Figure 21.4 shows a student working in the Robotic and Automation Laboratory for controlling a tridimensional digitizer within a scientific initiation program.



Figure 21.4. *Student activities in a scientific initiation program*

Again teachers play an important role because they coordinate activities and seek scholarships for the students. Coordinators of the course always try to show the students the importance of participating in this type of activity, which regularly has lectures by professors and scholarship students. This type of activity has been a reference for students even after graduation because research depends on the initiative necessary to solve problems and to learn this in an autodidactic manner.

21.3.4. *Participation in Junior Companies*

The META junior company of the School of Mechanical Engineering FEMEC (<http://www.meta.mecanica.ufu.br/>) was founded in 1992 and is one of the most reputable junior companies at UFU. In the 20 years of its existence, META has developed various projects, mainly in the development and improvement of products and processes, serving small, medium and big businesses.

The META junior company has been active in Junior Brazil, the Brazilian Confederation of Junior Enterprises, which represents junior companies nationwide and develops the Junior Enterprise Movement as an agent of business education and of new business generation ([http:// www.brasiljunior.org.br](http://www.brasiljunior.org.br)).

21.3.5. *Academic mobility*

Academic mobility can exist between the federal institutions of higher education in Brazil, or between foreign educational institutions that have agreements with the UFU.

The Federal University of Uberlândia has a long history of international scientific-technical cooperation, which began in 1987 with the *Institut National des Sciences Appliquées de Lyon INSA Lyon*, France, through an exchange agreement between Brazilian and French undergraduate students in mechanical engineering.

In recent years, the Federal University of Uberlândia has greatly increased scientific-technical cooperation with foreign institutions. Today, the University has 85 bilateral agreements with international institutions. Seven of these offer double degrees. The majority of these agreements, 55, are with Europe. Of these 55 agreements, 30 are with France.

Currently, with the incentive of the Brazilian government through its program “Science without Borders”, new opportunities have opened up, and countries that did not previously have a tradition of exchange programs with Brazil began offering opportunities for the exchange of undergraduate students. With these new opportunities, many students will have a chance to participate in exchange programs, to acquire knowledge and especially to have experience of other cultures, thus enriching their professional formation, which is essential in a globalized world.

To apply for scholarships in the international mobility programs, students must have a good academic performance and generally have participated in extracurricular activities such as scientific initiation and monitoring, and have experience in competition teams, etc.

From 2007 until the end of 2010, 30 mechatronics engineering students have participated in international mobility. In 2011, six students returned for the second half of the 2012 semester, and 17 left Brazil to participate in international mobility. This increase in participation is related to the availability of scholarships through the “Science without Borders” program, and the number of mechatronic students participating in the program is very expressive when compared to that of the other courses at the Federal University of Uberlândia.

21.3.6. Participation in competition teams

Participation in competitions is a form of entertainment combined with the application of technical and scientific knowledge. Besides this, the students learn by active participation in the event, because many of them have not attended the necessary specific disciplines required by the specific activity.

Students are introduced to this type of activity from the first period of the “Introduction to Mechatronics Engineering Course”, where they are divided into groups of three or four students. These groups have to develop a simple engineering project for a competition at the end of the semester. To develop this system, students must use basic course content, studied throughout the semester, or knowledge acquired even before entering the University. Besides the competition, each team has to write a report on the project, and justify the adopted solution with a subsequent oral presentation.

During the course, the organized teams focused on the competition are described as follows: (1) the Tucano Aero Design Team (<http://www.mecanica.ufu.br/node/193>), which aims to develop prototypes for model aircraft competition at SAE Brazil Aero Design (http://www.saebrasil.org.br/eventos/programas_estudantis). As per the rules of the competition, projects are judged according to several criteria: preparation of reports, technical drawings, an oral presentation of the project, maximum weight to be transported during test flights and accuracy in predicting the maximum transportable weight. (2) the Mini-Baja Cerrado Team, established in 1998, aims at the development and construction of recreational vehicles with four wheels. One of these participates annually in a competition sponsored by SAE Brazil. The following are evaluated in each competition: project reports, cost of mass production, safety aspects, operating comfort, traction capacity, steep ramp capacity, acceleration, top speed, braking, maneuverability and durability. (3) The EDROM Team for the Development of Mobile Robotics aims to participate in competitions involving the development of mechatronic systems geared toward mobile robotics. The team has participated in national competitions and in the South American Cup in Humanoid Robot categories (<http://www.robocup.org>) and SEK IEEE Standard Educational Kits (<http://www.cbr2011.org/sek.htm>) and (<http://ewh.ieee.org/reg/9/robotica>). Figure 21.5 shows the EDROM students in their workplace.



Figure 21.5. *The workplace of the EDROM students*

The teachers responsible for orientating and coordinating the work have a decisive role in the success of the teams.

21.3.7. Participation in student directories

The participation of students in the academic directories at UFU is much more associated with the political aspect than with the technical training. Students who take part in the academic directories have the opportunity to participate in the higher councils of the University, and to participate in philosophical discussions about engineering and about their educational institution. These discussions also take into account the relationships and influences that students have on society. All this is extremely enriching for the formation of responsible citizens.

The mechatronics engineering course has its own student directory with a vision more focused on the quality of the course and on the formation of students.

21.3.8. Participation and organization of events

The students can participate and organize scientific and/or academic events. For example, each year the students of mechatronic, mechanical and aeronautical engineering organize a scientific-technical event called the Week of Mechanical, Mechatronic and Aeronautical Engineering (SEMEC) that has celebrated its 14th annual event in 2012, which includes seminars, short courses, technical presentations of works by undergraduate and postgraduate students. The target audience is the academic community, businesses professionals in the city and in the region, teachers and engineering technicians in general.

21.3.9. Other activities

In addition to the activities listed above, students can participate in a special program, created and funded by the Ministry of Education in 1979, called the Tutorial Education Program (PET). PET groups are formed of 12 students, guided by a tutor who is responsible for the orientation, coordination and good performance of the group.

The following are some PET characteristics: ample interdisciplinary academic formation, collective action, continuous interaction, and planning and the executing of a diverse program of cultural and scientific activities.

The PET program at the Faculty of Mechanical Engineering (PETMEC) was established in 1992 and has actively participated in the formation of professionals in engineering.

21.4. Discussions

The accumulation of theoretical content at the beginning of engineering courses discourages the students, and has, as a consequence, the abandonment of the course by the students in the first years. When the students have the opportunity to get involved with extracurricular practical activities early in the course, they get motivated and acquire an overview regarding the application of the theoretical content, and this helps them to overcome the difficulties encountered in the first semesters.

The students who get involved with practical activities have experience of real problems before they see the theory referring to them, and this stimulates their interest in the theoretical content, because they can see the application of theory when it is taught in the classroom.

The globalized world requires a great versatility in engineering professionals and, in this case, multidisciplinary courses such as mechatronics engineering require that young engineers have a much broader view of systems, and also know how to tackle practical problems. The practical activities during the course make the mechatronic engineer distinctive professionals. Because of the large field of performance, mechatronics engineering students can get involved in all the extracurricular activities offered at the Federal University of Uberlândia, so they can start their practical activities in the first periods of the course.

Biannual meetings of the course collegiate encourage the students to participate in extracurricular activities from the beginning of the course. Graduated students are also invited to present their vision of the educational process that they experienced during the course.

All the extracurricular activities in which students participate can be integrated in the curriculum, and have a value equivalent to 90 h in the mechatronics engineering course at the Federal University of Uberlândia.

The importance of the commitment of teachers in the process of professional formation should be stressed. This is because it is teachers who look for scholarships, coordinate research and service projects and

competition teams. For this reason, it is necessary that teachers be motivated and be aware of their importance for the course and student formation.

The collegiate of the course undertakes semi-annual monitoring of the academic performance of mechatronics engineering students. From these analyzes, it has been possible to identify potential adjustments in the curriculum in order to achieve greater efficiency in learning.

Early in the course, some disciplines had laboratory activities as a complement to the teaching activities. However, it can be verified from other disciplines with laboratory activities, in which students participate effectively in practical activities, that increasing the laboratory workload could improve the quality of learning and thus academic student performance.

The following tables present statistical data since the beginning of the course in mechatronics engineering at the Federal University of Uberlândia. Table 21.1. shows examples of the approval rating of disciplines that have practical classes with effective participation of students in all of the planned activities, and Table 21.2 shows the approval rating for some disciplines that have no practical classes. From the analysis, it was possible to include some practical classes to enable effective participation of students in laboratory activities. No statistical data are shown because the changes were implemented only for two semesters. It is worth mentioning that we recall effective participation of students in laboratory activities in which the student develops the system proposed by the teacher. In this case, the teacher acts as a supervisor of the realized work.

Discipline	Approval rating (%)
Applied programming to engineering	91.1
Computer-aided design	93.1
Industrial automation	95.7
Industrial network	94.7
Pneumatic and hydraulic command systems	93.3

Table 21.1. *Approval rating in disciplines with effective participation of students in laboratory activities*

Discipline	Approval rating (%)
Digital control of systems	64.7
Digital electronics	81.7
Electrical circuits for mechatronic	75.2
Kinematics	74.9
Numerical calculation	77.1
Principle of materials science	74.6
Thermodynamics	73.8

Table 21.2. *Approval rating of students in disciplines without laboratory activities*

Another important aspect to be noted is that all students who received scholarship for international mobility, a total of 104, since the beginning of the course, have already participated in practical activities (sections 21.3.2–21.3.9), beyond the activities of compulsory disciplines. In 2012, from 274 students on the course, 25 students went abroad through the international mobility program.

In terms of incentives for students to participate in extracurricular activities and of their monitoring by the collegiate of the course, what can be seen from the beginning of the undergraduate mechatronics engineering course at the Federal University of Uberlândia is that dropout is small, the students are very well qualified and get good professional placements.

21.5. Conclusions

Encouraging the participation of students in extracurricular practice activities in the early periods is of a fundamental importance for mechatronics engineering students. This type of participation gives the students a vision of where theory can be applied. Thus, the students have a share in the construction and contextualization of the theory given in the classroom.

The involvement of students in extracurricular activities such as research initiation activities, PET, Mini-Baja, EDROM, and Aero Design, makes them feel more motivated, and able to see and experience real engineering problems in a safe manner.

Lack of student motivation is one of the most important reasons for the high number of dropouts from engineering courses, and involvement from the beginning of the course in practical activities can reduce this number in the early periods of the course.

This process of encouragement and participation in extracurricular practical activities needs the decisive participation of teachers, because it is they who coordinate all the activities, from the obtaining of scholarships to the formation of the young engineers.

21.6. Bibliography

- [BAR 04] BARBOSA C.S., PENNO E.J., OLIVEIRA V.F., “Integração e Contextualização de Conhecimentos nos Cursos de Engenharia”, *XXIV Encontro Nac. de Eng. de Produção*, Florianópolis, 2004.
- [BAZ 00] BAZZO W.A., PEREIRA L.T.V., LINSINGEN I.V., *Educação Tecnológica: Enfoques para o Ensino de Engenharia*, Editora da UFSC, Florianópolis, 2000.
- [BEV 09] BEVILACQUA L., “The university in times of cultural shock”, *20th International Congress of Mechanical Engineering – COBEM 2009*, Gramado, 2009.
- [BRI 93] BRINGUENTI I., *O Ensino de Engenharia na Escola Politécnica da USP: Fundamentos para o Ensino de Engenharia*, Ed. EPUSP, 1993.
- [CAR 03] CARVALHO J.C.M., TEODORO E.B., LACERDA H.B., *et al.*, Projeto Político Pedagógico do Curso de Graduação em Engenharia Mecatrônica (Political and Pedagogical Project for the undergraduate degree in Mechatronic Engineering), 2003.
- [CAR 11] CARVALHO J.C.M., “The role of ABCM in Engineering and Mechanical Sciences in Brazil and its relationship with IFToMM”, in CECCARELLI M. (eds), *Technology Developments: The Role of Mechanism and Machine Science and IFToMM*, Springer, 2011.
- [FEL 11] FELIPES B.A., AGUIAR J.G., DINIZ A.C.G.C., “Sistema da Qualidade em Laboratórios Universitários: Incentivo ao Ensino, Pesquisa e Extensão”, *Revista de Ensino de Engenharia*, vol. 30, no. 2, pp. 14–23, 2011.
- [GUI 12] GUIZZO E., “Engineering schools that tie theory and practice together retain ore Students”, *Spectrum IEEE Pesquisa*, available at http://spectrum.ieee.org/tech-talk/semiconductors/devices/engineering_schools_that_tie_t (accessed on 15 August 2012).

- [HAB 06] HABIB M.K., “Mechatronics engineering the evolution, the needs and the challenges”, *Proceedings of the 32nd Annual Conference of IEEE Industrial Electronics Society, ECON 2006*, IEEE, pp. 4510–4515, 2006.
- [HAB 07] HABIB M.K., “Mechatronics: a unifying interdisciplinary and intelligent Engineering paradigm”, *IEEE Industrial Electronics Magazine*, IEEE, vol. 1, no. 2, pp. 12–24, 2007.
- [HOW 09] HOWELL J.R., “The cloudy crystal ball and engineering education”, *20th International Congress of Mechanical Engineering – COBEM 2009*, Gramado, 2009.
- [MAN 12] MANESTRINA T.C., <http://revistas.udesc.br/index.php/udescvirtual/article/viewFile/1658/1333-20/07/2012>, 2012.
- [PET 94] PETITAT A., *Produção da Escola/Produção da Sociedade*, Artes Médicas, Porto Alegre, 1994.
- [VEI 95] VEIGA I.P.A. (org.), *Projeto Político-Pedagógico da Escola: Uma Construção Possível*, Papirus, Campinas, SP, 1995.

List of Authors

Shinya AOI
Department of Aeronautics
and Astronautics
Graduate School of Engineering
Kyoto University
Japan

Yoshio AOKI
College of Science and
Technology
Nihon University
Chiba
Japan

Evangelos BOUKAS
Robotics and Automation
Laboratory
Department of Production and
Management Engineering
Democritus University of Thrace
Xanthi
Greece

João Carlos M. CARVALHO
School of Mechanical
Engineering
Federal University of Uberlândia
Brazil

Matteo CIANCHETTI
The BioRobotics Institute
Scuola Superiore Sant'Anna
Pisa
Italy

Faruk DAUTOVIC
Faculty of Electrical Engineering
Zmaja od Bosne bb
Sarajevo
Bosnia and Herzegovina

J. Paulo DAVIM
University of Aveiro
Portugal

Raffaele DI GREGORIO
Department of Engineering
University of Ferrara
Italy

Duygun EROL BARKANA
Department of Electrical
and Electronics Engineering
Yeditepe University
Istanbul
Turkey

Gian Luca FORESTI
Department of Mathematics and
Computer Science (DIMI)
University of Udine
Italy

Vera Lúcia D.S. FRANCO
School of Mechanical Engineering
Federal University of Uberlândia
Brazil

Toshio FUKUDA
Department of Micro-Nano
Systems Engineering and Center
for Micro-Nano Mechatronics
Nagoya University
Aichi
Japan

Antonios GASTERATOS
Robotics and Automation
Laboratory
Department of Production and
Management Engineering
Democritus University of Thrace
Xanthi
Greece

Maki K. HABIB
Mechanical Engineering
Department
School of Sciences and
Engineering
The American University in Cairo
New Cairo
Egypt

Moharam HABIBNEJAD KORAYEM
School of Mechanical Engineering
Iran University of Science and
Technology
Tehran
Iran

Emiko HARA
College of Science and
Technology
Nihon University
Chiba
Japan

Yun HO TSOI
Faculty of Engineering
The University of Auckland
Auckland
New Zealand

Muhidin HUJDUR
Faculty of Electrical
Engineering
Zmaja od Bosne bb
Sarajevo
Bosnia and Herzegovina

Michal IRANI
Robotic Research Laboratory
Iran University of Science and
Technology
Tehran
Iran

M. KARIMI
Department of Mechanical
Engineering
K. N. Toosi University of
Technology
Tehran
Iran

Admir KAKNJO
Faculty of Electrical Engineering
Zmaja od Bosne bb
Sarajevo
Bosnia and Herzegovina

Ioannis KOSTAVELIS
Robotics and Automation
Laboratory
Department of Production and
Management Engineering
Democritus University of Thrace
Xanthi
Greece

Hongbo LI
Department of Computer Science
and Technology
Tsinghua University
Beijing
China

Kai LIU
Department of Computer Science
and Technology
Tsinghua University
Beijing
China

Soleiman MANTEGHI
Science and Research Branch
College of Computer and
Mechatronics Engineering
Islamic Azad University
Tehran
Iran

Niki MARTINEL
Department of Mathematics and
Computer Science (DIMI)
University of Udine
Italy

Andrew MCDAID
Faculty of Engineering
The University of Auckland
Auckland
New Zealand

Christian MICHELONI
Department of Mathematics and
Computer Science (DIMI)
University of Udine
Italy

S. Ali A. MOOSAVIAN
Department of Mechanical
Engineering
K. N. Toosi University of
Technology
Tehran
Iran

Masahiro NAKAJIMA
Center for Micro-Nano
Mechatronics
Nagoya University
Aichi
Japan

Lazaros NALPANTIDIS
Computer Vision and Machine
Intelligence Laboratory
Department of Mechanical and
Manufacturing Engineering
Aalborg University
Copenhagen
Denmark

Nedim OSMIC
Faculty of Electrical Engineering
Zmaja od Bosne bb
Sarajevo
Bosnia and Herzegovina

Fatih ÖZKUL
Figes Engineering
Istanbul
Turkey

Balan PILLAI
Aalto University
School of Engineering
Finland

Saeed RAFEE NEKOO
Robotic Research Laboratory
Iran University of Science and
Technology
Tehran
Iran

Gerasimos RIGATOS
Unit of Industrial Automation
Industrial Systems Institute
Rion Patras
Greece

Aleksandar RODIĆ
Mihajlo Pupin Institute
Robotics Laboratory
Belgrade
Serbia

Ken SAITO
Department of Precision
Machinery Engineering
College of Science and
Technology
Nihon University
Chiba
Japan

Vesa SALMINEN
Spiral Business Center Oy
Myyrmäentie
Vantaa
Finland

Yoshifumi SEKINE
Department of Electronics and
Computer Engineering
College of Science and
Technology
Nihon University
Chiba
Japan

Pierluigi SIANO
Department of Industrial
Engineering
University of Salerno
Fisciano
Italy

Ivan STOJKOVIĆ
Mihajlo Pupin Institute
Robotics Laboratory
Belgrade
Serbia

Zengqi SUN
Department of Computer Science
and Technology
Tsinghua University
Beijing
China

Akihisa TABATA
College of Science and
Technology
Nihon University
Chiba
Japan

Hiroataka TAJIMA
Department of Micro-Nano
Systems Engineering
Nagoya University
Aichi
Japan

Keiki TAKADAMA
The University of
Electro-Communications
Chofu
Tokyo
Japan

Minami TAKATO
Department of Precision
Machinery Engineering
College of Science and
Technology
Nihon University
Chiba
Japan

Kouki TAKEMORI
Nara National College of
Technology
Yamatokoriyama
Nara
Japan

Masaru TAKEUCHI
Department of Micro-Nano
Systems Engineering
Nagoya University
Aichi
Japan

Hami TOURAJIZADEH
School of Mechanical Engineering
Iran University of Science and
Technology
Tehran
Iran

Fumio UCHIKOBA
Department of Precision
Machinery Engineering
College of Science and
Technology
Nihon University
Chiba
Japan

Jasmin VELAGIC
Faculty of Electrical Engineering
Zmaja od Bosne bb
Sarajevo
Bosnia and Herzegovina

Marco VERNIER
Department of Mathematics and
Computer Science (DIMI)
University of Udine
Italy

Shengquan XIE
Faculty of Engineering
The University of Auckland
Auckland
New Zealand

Tomohiro YAMAGUCHI
Nara National College of
Technology
Yamatokoriyama
Nara
Japan

Tao YUE
Department of Micro-Nano
Systems Engineering
Nagoya University
Aichi
Japan

Nikolaos ZERVOS
Unit of Digital Communications
Industrial Systems Institute
Rion Patras
Greece

Index

A

Activation, 244–248
Actuator, 324–328
Actuator force control, 176–198
Actuator-sensor-environment,
182–195
Android, 453, 454, 455, 470, 475
ARM processor, 483–497
Axon model, 104

B

Biological cell assembly, 30–33
Boundary detector, 463–464

C

Cell body model, 99–104
Compass model, 41–43
Contact manipulation, 23–28
Continuous learning, 69–93
CPG-based locomotion control
system, 38–39

D

Dead reckoning, 373–375
Degrees of freedom (dof), 113, 139,
146, 389

Derivative-free, 277–279, 298–304,
308
Differential flatness, 278, 288–293
Discrete circuit, 101, 108

E

Encoder, 154, 316, 317, 323, 328–330
Ethernet, 392

F

Fixed based manipulator, 265–269
Flat system, 279, 288–293

G

Graphical user interface (GUI),
347–356
Group theory, 115, 120–122

H

Hall sensor, 223
Holbos, 423–447

I

Interlimb coordination, 60
Intralimb coordination, 39, 40

J

Jacobian algorithm, 259–265

K

Kinect, 434, 551, 564

Kinematic model, 426–430

Kinetostatic performance, 131–138

L

Lie algebra, 282–288

Limb, 113–124

M

Maze model, 75–80, 84, 85, 90–92

Memory effect, 228, 230, 240

MEMS, 6, 371, 552

Micro-nanomechatronics, 19–33

Micro-organism, 25

MIMO, 163–204

Mobile base manipulator,
269–272

Mobile device, 451–476

Modified transpose Jacobian (MTJ),
255, 257

MOEMS, 6

Multiple impedance control (MIC),
261

N

Non-contact manipulation, 23, 28

Nonlinear system, 282, 283, 288,
289, 292, 536

O

On-board controller, 389, 391, 440

P

Parallel mechanism, 171–175

Parallel robot, 163–204

Parallel wrist, 113–141

Permanent magnet synchronous
generator (PMSG), 277,
279–282

Phase regulation, 40

Phase resetting, 37–64

Photo-linkable resin, 26,
30–33

Product development, 513–525

Proportional-integral derivative
(PID), 149, 495

Pulse width modulation (PWM), 95,
323, 393, 440

Pulse-type hardware, 95–109

PWM servo motor, 97–98

R

RehabRoby, 145–159

Reusable, 91

RGB, 434

Robot-assisted rehabilitation system,
145–159

Rotor, 183, 280, 281, 299, 305, 308,
402, 403

S

Scanning laser, 369–382

Screw theory, 113, 115

Sensory regulation, 40–41

Shape memory alloys (SMA),
227–251

Signal processing, 6, 176, 217,
218–219, 371, 406

Single cell analysis, 20, 25, 26, 28,
30–33

Single machine infinite bus (SMIB),
279, 280

SLAM, 370

Superelastic, 228, 230–231

Synaptic model, 103

T

3D reconstruction, 554–556
Transpose Jacobian control (TJ),
255–272
Tuned modified transpose Jacobian
control (TMTJ), 255–272

U

Ultrasonic positioning system,
375–377, 379, 380
Unknown usage, 91, 92
Upper extremities, 146, 148

V

Virtual laboratory, 247–352

W

Wheel-based, 385–415
Wi-Fi, 389, 390, 392, 393