Helmut E. Graeb

Analog Design Centering and Sizing

Springer

Analog Design Centering and Sizing

ANALOG DESIGN CENTERING AND SIZING

HELMUT E. GRAEB
Institute for Electronic Design Automation
Technische Universitaet Muenchen, Germany

Springer

A C.I.P. Catalogue record for this book is available from the Library of Congress.

ISBN 978-1-4020-6003-8 (HB)
ISBN 978-1-4020-6004-5 (e-book)

Published by Springer,
P.O. Box 17, 3300 AA Dordrecht, The Netherlands.

www.springer.com

Printed on acid-free paper

Contents

List of Figures xi

List of Tables xix

Preface xxi

1. INTRODUCTION 1
   - 1.1 Integrated Circuits 1
   - 1.2 Analog Circuits 4
   - 1.3 Analog Design 6
   - 1.4 Analog Sizing 13
2. TOLERANCE DESIGN: EXAMPLE 19
   - 2.1 RC Circuit 19
   - 2.2 Performance Evaluation 20
   - 2.3 Performance-Specification Features 20
   - 2.4 Nominal Design/Performance Optimization 20
   - 2.5 Yield Optimization/Design Centering 22
3. PARAMETERS & TOLERANCES, PERFORMANCE & SPECIFICATION 27
   - 3.1 Parameters 27
   - 3.2 Parameter Tolerances 28
   - 3.3 Range-Parameter Tolerances 29
   - 3.4 Statistical Parameter Distribution 31
   - 3.5 Univariate Normal Distribution 32
   - 3.6 Multivariate Normal Distribution 34
   - 3.7 Transformation of Statistical Distributions 39
   - 3.8 Generation of Normally Distributed Sample Elements 43

3.9 Global and Local Parameter Tolerances 44

3.10 Performance Features 45

3.11 Numerical Simulation 46

3.12 Performance-Specification Features 47

4. ANALOG SIZING TASKS 49

4.1 Sensitivity-Based Analysis 49

4.1.1 Similarity of Performance Features 51

4.1.2 Similarity of Parameters 51

4.1.3 Significance of Parameters 51

4.1.4 Adjustability of Performance Features 52

4.1.5 Multiple-Objective Behavior 52

4.1.6 Exercise 52

4.1.7 Sensitivity-Based Analysis of Tolerance Objectives 53

4.2 Performance-Sensitivity Computation 53

4.2.1 Simulator-Internal Computation 53

4.2.2 Finite-Difference Approximation 53

4.3 Scaling Parameters and Performance Features 54

4.3.1 Scaling to a Reference Point 54

4.3.2 Scaling to the Covered Range of Values 54

4.3.3 Scaling by Affine Transformation 55

4.3.4 Scaling by Equalization of Sensitivities 55

4.4 Nominal Design 56

4.5 Multiple-Objective Optimization 56

4.5.1 Smaller or Greater Performance Vectors 57

4.5.2 Pareto Point 59

4.5.3 Pareto Front 59

4.5.4 Pareto Optimization 61

4.6 Single-Objective Optimization 61

4.6.1 Vector Norms 63

4.6.2 Performance Targets 63

4.7 Worst-Case Analysis and Optimization 64

4.7.1 Worst-Case Analysis 64

4.7.2 Worst-Case Optimization 68

4.8 Yield Analysis, Yield Optimization/Design Centering 70

4.8.1 Yield 70

4.8.2 Acceptance Region Partitions 72

4.8.3 Yield Partitions 75

4.8.4 Yield Analysis 75
4.8.5 Yield Optimization/Design Centering 76
4.8.6 Tolerance Assignment 80
4.8.7 Beyond 99.9% Yield 81

5. WORST-CASE ANALYSIS 85
5.1 Classical Worst-Case Analysis 86
5.1.1 Classical Worst-Case Parameter Vectors 88
5.1.2 Classical Worst-Case Performance Values 89
5.1.3 Discrete Parameters 89
5.1.4 Corner Worst Case 90
5.2 Realistic Worst-Case Analysis 90
5.2.1 Realistic Worst-Case Parameter Vectors 93
5.2.2 Realistic Worst-Case Performance Values 93
5.3 Yield/Worst-Case Distance – Linear Performance Feature 93
5.4 General Worst-Case Analysis 95
5.4.1 General Worst-Case Parameter Vectors 98
5.4.2 General Worst-Case Performance Values 98
5.5 Yield/Worst-Case Distance – Nonlinear Performance Feature 99
5.5.1 Yield Approximation Accuracy 100
5.5.2 Realistic Worst-Case Analysis as Special Case 105
5.6 Exercise 105

6. YIELD ANALYSIS 107
6.1 Statistical Yield Analysis 107
6.1.1 Monte-Carlo Analysis 109
6.1.2 Importance Sampling 110
6.1.3 Yield Estimation Accuracy 111
6.2 Tolerance Classes 115
6.2.1 Tolerance Interval 115
6.2.2 Tolerance Box 116
6.2.3 Tolerance Ellipsoid 118
6.2.4 Single-Plane-Bounded Tolerance Region 121
6.2.5 Corner Worst Case vs. Realistic Worst Case 126
6.3 Geometric Yield Analysis 128
6.3.1 Problem Formulation 128
6.3.2 Lagrangian Function 131
6.3.3 First-Order Optimality Condition 132
6.3.4 Second-Order Optimality Condition 133

6.3.5 Worst-Case Range-Parameter Vector 135
6.3.6 Worst-Case Statistical Parameter Vector 135
6.3.7 Worst-Case Distance 136
6.3.8 Geometric Yield Partition 137
6.3.9 Geometric Yield 138
6.3.10 General Worst-Case Analysis/Geometric Yield Analysis 141
6.3.11 Approximate Geometric Yield Analysis 142
6.4 Exercise 143
7. YIELD OPTIMIZATION/DESIGN CENTERING 145
7.1 Statistical-Yield Optimization 145
7.1.1 Acceptance-Truncated Distribution 145
7.1.2 Statistical Yield Gradient 146
7.1.3 Statistical Yield Hessian 148
7.1.4 Solution Approach to Statistical-Yield Optimization 149
7.1.5 Tolerance Assignment 150
7.1.6 Deterministic Design Parameters 150
7.2 Geometric-Yield Optimization 153
7.2.1 Worst-Case-Distance Gradient 153
7.2.2 Solution Approaches to Geometric-Yield Optimization 156
7.2.3 Least-Squares/Trust-Region Solution Approach 157
7.2.4 Min-Max Solution Approach 162
7.2.5 Linear-Programming Solution Approach 162
7.2.6 Tolerance Assignment, Other Optimization Parameters 163

Appendices

A Expectation Values 165
A.1 Expectation Value 165
A.2 Moments 165
A.3 Mean Value 165
A.4 Central Moments 166
A.5 Variance 166
A.6 Covariance 166
A.7 Correlation 166
A.8 Variance/Covariance Matrix 166
A.9 Calculation Formulas 167
A.10 Standardization of Random Variables 168

A.11 Exercises 168
B Statistical Estimation of Expectation Values 169
B.1 Expectation-Value Estimator 169
B.2 Variance Estimator 169
B.3 Estimator Bias 170
B.4 Estimator Variance 170
B.5 Expectation-Value-Estimator Variance 170
B.6 Estimator Calculation Formulas 171
B.7 Exercises 171
C Optimality Conditions of Nonlinear Optimization Problems 173
C.1 Unconstrained Optimization 173
C.2 First-Order Unconstrained Optimality Condition 174
C.3 Second-Order Unconstrained Optimality Condition 175
C.4 Constrained Optimization 176
C.5 First-Order Constrained Optimality Condition 177
C.6 Second-Order Constrained Optimality Condition 179
C.6.1 Lagrange-Factor and Sensitivity to Constraint 180
C.7 Bounding-Box-of-Ellipsoids Property (37) 181
References 183
Index 193

List of Figures

1 Trends in process and design technology. Data have been taken from the Intel web site and from the International Technology Roadmap for Semiconductors (ITRS) web site. 2

2 Mixed-signal system-on-chip ICs. 3

3 A CMOS operational transconductance amplifier (OTA, left) and a CMOS folded-cascode operational amplifier (right). 4

4 An OTA-C biquad filter with OTAs from Figure 3 as building blocks [57]. 5

5 A phase-locked loop (PLL) with digital components, phase frequency detector (PFD), divider (DIV), and analog components, charge pump (CP), loop filter (LF), voltage-controlled oscillator (VCO). 5

6 Scope of analog design. 6

7 Nonlinear analog circuit. Structural model in form of a circuit netlist, behavioral model in form of a differential equation. Concerning the output voltage, both models are equivalent. 8

8 Simplified models of the OTA in Figure 3. Behavioral model in form of a description language code, Structural model in form of a circuit netlist. Concerning the OTA ports, both models are equivalent and approximate to the transistor netlist in Figure 3. 9

9 Analog synthesis and analysis. 9

10 Design flow of a PLL. As in Figure 9, boxes denote the structural view, rounded boxes the behavioral view. 11

11 Analog synthesis path of a PLL. 12

12 Analog sizing tasks. 13

13 Tolerance design tasks. 14

14 Elementary RC circuit with performance features time constant τ and area A as a function of resistor value R and capacitor value C. 19

15 RC circuit. *(a)* Performance values of RC circuit after nominal design. *(b)* Parameter values of RC circuit after nominal design. 21

16 Parameter values and parameter tolerances of the RC circuit after nominal design. 23

17 Probability density function after nominal design of the RC circuit, truncated by the performance specification. 24

18 *(left)* Parameter values and truncated probability density function. after nominal design of the RC circuit. *(right)* Parameter values and truncated probability density function after yield optimization/design centering of the RC circuit. 25

19 *(a)* Box tolerance region. *(b)* Polytope tolerance region. *(c)* Ellipsoid tolerance region. *(d)* Nonlinear tolerance region. 30

20 Probability density function pdf and cumulative distribution function cdf of a univariate normal distribution. 34

21 Multivariate normal distribution for two parameters according to (23), (24) or (34), respectively, with mean values $x_{s,0,1} = x_{s,0,2} = 0$, correlation $\varrho = 0.5$, and variances $\sigma_1 = 2$, $\sigma_2 = 1$. 35

22 Different sets of level contours $\beta^2 = (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0})$ (24) of a two-dimensional normal probability density function pdf_N. *(a)* Varying β, constant positive correlation ϱ, constant unequal variances σ_1, σ_2. *(b)* Varying β, constant zero correlation, constant unequal variances. *(c)* Varying β, constant zero correlation, constant equal variances. *(d)* Varying correlation, constant β, constant unequal variances. 38

23 Sensitivity matrix in the parameter and performance space. 50

24 *(a)* Set $M^{>}(\mathbf{f}^*)$ of all performance vectors that are greater than $\mathbf{f}^*$, i.e. inferior to $\mathbf{f}^*$ with regard to multiple-criteria minimization according to (92) and (93). *(b)* Set $M^{<}(\mathbf{f}^*)$ of all performance vectors that are less than $\mathbf{f}^*$, i.e. superior to $\mathbf{f}^*$ with regard to multiple-criteria minimization. 58

25 *(a)* Pareto front $PF(f_1, f_2)$ of a feasible performance region of two performance features. Different Pareto points addressed through different weight vectors from reference point $\mathbf{f}_{RP}$, which is determined by the individual minima of the performance features. *(b)* Pareto front $PF(f_1, f_2, f_3)$ of three performance features with boundary $\mathcal{B} = \{PF(f_1, f_2), PF(f_1, f_3), PF(f_2, f_3)\}$. 60

26 *(a)* Continuous Pareto front of a convex feasible performance region. $f^*_{I,2}$ is monotone in $f^*_{I,1}$. *(b)* Continuous Pareto front of a nonconvex feasible performance region. $f^*_{I,2}$ is monotone in $f^*_{I,1}$. *(c)* Discontinuous Pareto front of a nonconvex feasible performance region. $f^*_{I,2}$ is nonmonotone in $f^*_{I,1}$. 61

27 *(a)* Level contours of the weighted l_1-norm. *(b)* Level contours of the weighted l_2-norm. *(c)* Level contours of the weighted l_∞-norm. 63

28 Input and output of a worst-case analysis. 66

29 *(a)* Input of a worst-case analysis in the parameter and performance space: nominal parameter vector, tolerance regions. *(b)* Output of a worst-case analysis in the parameter and performance space: worst-case parameter vectors, worst-case performance values. 67

30 Nested loops within worst-case optimization. 69

31 *(a)* The volume under a probability density function of statistical parameters, which has ellipsoid equidensity contours, corresponds to 100% yield. *(b)* The performance specification defines the acceptance region A_f, i.e. the region of performance values of circuits that are in full working order. The yield is the portion of circuits in full working order. It is determined by the volume under the probability density function truncated by the corresponding parameter acceptance region A_s. 71

32 Parameter acceptance region A_s partitioned into parameter acceptance region partitions, $A_{s,L,1}$, $A_{s,U,1}$, $A_{s,L,2}$, $A_{s,U,2}$, for four performance-specification features, $f_1 \geq f_{L,1}$, $f_1 \leq f_{U,1}$, $f_2 \geq f_{L,2}$, $f_2 \leq f_{U,2}$. A_s results from the intersection of the parameter acceptance region partitions. 74

33 Input and output of a yield analysis. 76

34 Yield optimization/design centering determines a selected point of Pareto front of performance features. 77

35 Nested loops within yield optimization/design centering. 78

36 *(a)* Initial situation of yield optimization/design centering by tuning of design parameters $\mathbf{x}_d$ that are disjunct from statistical parameters. *(b)* After yield optimization/design centering by tuning of design parameters $\mathbf{x}_d$ that are disjunct from statistical parameters. The parameter acceptance region A_s depends on the values of design parameters $\mathbf{x}_d$. The equidensity contours of a normal probability density function are ellipsoids according to (24). 79

37 *(a)* Yield optimization/design centering by tuning of the mean values of statistical parameters $\mathbf{x}_{s,0}$. Parameter acceptance region A_s is constant. *(b)* Yield optimization/design centering by tuning of the mean values, variances and correlations of statistical parameters $\mathbf{x}_{s,0}$, $\mathbf{C}$ (tolerance assignment). Parameter acceptance region A_s is constant. Level contours of the normal probability density function (24) change their shape due to changes in the covariance matrix $\mathbf{C}$. 80

38 To go beyond 99.9% yield for maximum robustness, yield optimization/design centering requires specific measures. 82

39 Classical worst-case analysis. 87

40 Classical worst-case analysis with undefined elements $x_{r,WL,1}$, $x_{r,WU,1}$, of worst-case parameter vectors $\mathbf{x}_{r,WL}$, $\mathbf{x}_{r,WU}$. 89

41 Normal probability density function of a manufactured component splits into truncated probability density functions after test according to different quality classes. 90

42 Realistic worst-case analysis. 91

43 General worst-case analysis, in this case the solution is on the border of the tolerance region. 96

44 *(a)* Comparison between true parameter acceptance region partitions (gray areas) and approximate parameter acceptance region partitions (linen-pattern-filled areas) of a general worst-case analysis of an upper worst-case performance value. *(b)* Lower worst-case performance value. *(c)* Lower and upper worst-case performance value. 102

45 Duality principle in minimum norm problems. Shown are two acceptance regions (gray), the respective nominal points within acceptance regions and two points on the border of each acceptance region. Points (a) and (d) are the worst-case points. 103

46 Statistical yield estimation (Monte-Carlo analysis) consists of generating a sample according to the underlying statistical distribution, simulating of each sample element and flagging of elements satisfying the performance specification. 109

47 Tolerance box $T_{s,B}$ and normal probability density function $\mathrm{pdf}_{N,0R}$ with zero mean and unity variance. 117

48 Ellipsoid tolerance region $T_{s,E}$ and equidensity contours of normal probability density function pdf_N. 119

49 Single-plane-bounded tolerance region $T_{s,SP,U}$ and equidensity contours of normal probability density function pdf_N. 122

50 Single-plane-bounded tolerance regions $T_{s,SP,L}$, $T_{s,SP,U}$ for a single performance feature with either an upper bound (first row) or a lower bound (second row), which is either satisfied (first column) or violated at the nominal parameter vector (second column). 124

51 Single-plane-bounded tolerance region $T_{s,SP}$ with corresponding worst-case parameter vector $\mathbf{x}_{W,SP}$ and worst-case distance β_W for two parameters. β_W denotes a $\pm\beta_W$ times the covariances tolerance ellipsoid. Tolerance box $T_{s,B}$ determined by $\pm\beta_W$ times the covariances with corresponding worst-case parameter vector $\mathbf{x}_{W,B}$ and worst-case distance $\beta_{W,B}$. 126

52 Geometric yield analysis for a lower bound on a performance feature, $f > f_L$, which is satisfied at the nominal parameter vector. Worst-case parameter vector $\mathbf{x}_{s,WL}$ has the smallest distance from the nominal statistical parameter vector $\mathbf{x}_{s,0}$ measured according to the equidensity contours among all parameter vectors outside of or on the border of the acceptance region partition $A_{s,L}$. The tangential plane to the tolerance ellipsoid through $\mathbf{x}_{s,WL}$ as well as to the border of $A_{s,L}$ at $\mathbf{x}_{s,WL}$ determines a single-plane-bounded tolerance region $A'_{s,L}$. 129

53 Two stationary points $\mathbf{x}_{s,A}$ and $\mathbf{x}_{s,W}$ of (277), which satisfy the first-order optimality condition. Only $\mathbf{x}_{s,W}$ satisfies the second-order optimality condition and is therefore a solution of (270). 134

54 Worst-case distances and approximate yield values from a geometric yield analysis of the operational amplifier from Table 14. 139

55 Parameter acceptance region A_s (gray area) originating from four performance-specification features, $f_1 \geq f_{L,1}$, $f_1 \leq f_{U,1}$, $f_2 \geq f_{L,2}$, $f_2 \leq f_{U,2}$ (Figure 32). A geometric yield analysis leads to four worst-case parameter vectors $\mathbf{x}_{WL,1}$, $\mathbf{x}_{WU,1}$, $\mathbf{x}_{WL,2}$, $\mathbf{x}_{WU,2}$ and four single-plane-bounded tolerance regions. The intersection of these single-plane-bounded tolerance regions forms the approximate parameter acceptance region A'_s (linen-pattern-filled area). 140

56 General worst-case analysis and geometric yield analysis as inverse mappings exchanging input and output. 142

57 *(a)* Statistical-yield optimization before having reached the optimum. Center of gravity $\mathbf{x}_{s,0,\delta}$ of the probability density function truncated due to the performance specification (remaining parts drawn as bold line) differs from the center of gravity of the original probability density function. A nonzero yield gradient $\nabla Y(\mathbf{x}_{s,0})$ results. *(b)* After statistical-yield optimization having reached the optimum. Centers of gravity of original and truncated probability density function are identical. 148

58 Worst-case distances and approximate yield partition values before and after geometric-yield optimization of the operational amplifier from Table 15. 158

59 Least-squares/trust-region approach to geometric-yield optimization according to (366). Quarter-circles represent trust-regions of a step. Ellipses represent level contours of the least-squares optimization objective (363). $\mathbf{r}^*$ is the optimal step for the respective trust region determined by Δ. 160

60 Pareto front of optimal solutions of least-squares/trust-region approach (366) in dependence of maximum step length Δ, i.e. Lagrange factor λ (367). A point in the bend corresponds to a step with a grand progress towards the target worst-case distances at a small step length. 161

61 Min-max solution to geometric-yield optimization. 163

C1 Descent directions and steepest descent direction of a function of two parameters. 174

C2 Different types of definiteness of the second-derivative of a function of two parameters: positive definite (upper left), positive semidefinite (upper right), indefinite (lower left), negative definite (lower right). 176

C3 Descent directions and unconstrained directions of a function of two parameters with one active constraint. 178

List of Tables

1 Analog design terms. 7

2 Some modeling approaches for analog circuits in behavioral and structural view. 8

3 Characteristics of numerical optimization. 15

4 Sizing of an operational amplifier. 17

5 Selected cumulative distribution function values of the univariate normal distribution. 33

6 Worst-case analysis (WCA) types and characterization. 85

7 Standard deviation of the yield estimator if the yield is 85% for different sample sizes evaluated according to (225). 113

8 Required sample size for an estimation of a yield of 85% for different confidence intervals and confidence levels according to (232). 114

9 Tolerance intervals $T_{s,I}$ and corresponding yield values Y_I according to (235). 116

10 Yield values Y_B for a tolerance box $T_{s,B}$ with $\forall_k\ \beta_{a_k} = -\beta_{b_k} = -\beta_W = -3$ in dependence of the number of parameters n_{xs} and of the correlation. $\forall_{k \neq l}\ \varrho_{k,l} = 0.0$ according to (241), $\forall_{k \neq l}\ \varrho_{k,l} = 0.8$ according to (237). 118

11 Yield values Y_E for an ellipsoid tolerance region $T_{s,E}$ with $\beta_W = 3$ in dependence of the number of parameters n_{xs} according to (247) and (248). 120

12 Single-plane-bounded tolerance region $T_{s,SP}$ and corresponding yield partition values Y_{SP} according to (258) or (259). 125

13 Exaggerated robustness $\beta_{W,B}$ represented by a corner worst-case parameter vector of a classical worst-case analysis, for different correlation $\forall_{k \neq l} \varrho_{k,l} = \varrho$ among the parameters, and for different numbers of parameters n_{xs}. 127

14 Geometric yield analysis of an operational amplifier. 139

15 Geometric-yield optimization of an operational amplifier from Section 6.3.9. 157

Preface

This book represents a compendium of fundamental problem formulations of analog design centering and sizing. It provides a differentiated knowledge about the tasks of analog design centering and sizing. In particular the worst-case problem will be formulated. It stands at the interface between process technology and design technology. This book wants to point out that and how both process and design technology are required for its solution. Algorithms based on the presented material are for instance available in the EDA tool WiCkeD [88].

The intention is to enable analog and mixed-signal designers to assess CAD solution methods that are presented to them. On the other side, the intention is to enable developers of analog CAD tools to formulate and develop solution approaches for analog design centering and sizing. The structure of the book is geared towards a combination of a reference book and a textbook. The presentation goes from general topics to the more specific details, preceding material is usually a prerequisite for succeeding material. The formulations of tasks and solution approaches by mathematical means makes the book suitable as well for students dealing with analog design and design methodology.

The contents is structured as follows:

Chapter 1 sketches the role of analog circuits and analog design in integrated circuits. An overview of analog sizing tasks and the corresponding terminology is introduced.

Chapter 2 illustrates analog sizing and yield optimization/design centering with the simplest example of an RC circuit.

Chapter 3 describes the basic input and output quantities of analog sizing. Parameters and performance features are defined. Tolerance ranges and statistical distributions of parameters as well as the performance specification are introduced. The multivariate normal distribution as most important distribution type is described in the univariate and multivariatecase. In addition, the transformation of one type of distribution into another is sketched and illustrated with examples like that of the generation of a normally distributed sample element.

Chapter 4 formulates the basic tasks of analog sizing. The first task is a sensitivity-based analysis of the effects of multiple parameters and multiple design objectives. The second task is the scaling of parameters and design objectives, which is crucial for the effectiveness of sizing algorithms. Performance optimization is introduced as a multiple-objective approach, which leads to Pareto optimization, and as a single-objective approach using vector norms. After that, worst-case analysis and optimization are formulated and characterized. In the same way, yield analysis and optimization are treated. It is pointed out how yield and worst-case refer to each other, how yield optimization/design centering works beyond 99.9% yield, and how different quantities contribute to the yield.

Chapter 5 develops three types of worst-case analysis, which are called classical, realistic and general. Solution approaches for the three types of worst-case analysis are developed. In the case of classical and realistic worst-case analysis, analytical solutions are obtained. In a general worst-case analysis, a numerical optimization problem is formulated as a starting point for the development of solution algorithms. A general approach is described that relates a yield requirement to the tolerance input of a worst-case analysis.

Chapter 6 describes two types of yield analysis. The first type is a statistical estimation based on a Monte-Carlo analysis, the second type is a geometric approximation that is closely related to worst-case analysis. Advantages and limitations of both types are treated.

Chapter 7 develops two types of yield optimization/design centering. They are based on the two types of yield analysis described in Chapter 6 and have corresponding advantages and limitations. The problem formulations and approaches for solution algorithms are developed. Derivatives of the objective functions of yield optimization/design centering will be derived for sensitivity-based optimization approaches.

It is recommended that the reader is familiar with matrix/vector notations and the concept of optimality conditions. Appendices A-C summarize statistical expectation values and their estimation as well as optimality conditions of nonlinear optimization.

Chapter 1

INTRODUCTION

1.1 Integrated Circuits

Integrated circuits (ICs), so-called chips, have become the brain and nervous system of all kinds of devices. ICs are involved in everybody's everyday life more and more comprehensively. They are parts of domestic appliances, individual or public transportation devices like cars and trains, traffic management systems, communication networks, mobile phones, power plants and supply networks in high and low voltage, medical devices outside or inside the body, clothes, computers, and so on.

ICs are parts of systems, but ICs more and more form complete systems with hardware and software components in themselves. What had been an electronic cubicle with many components as tall as a human being years ago, today can be implemented on one IC as large as a thumb nail.

This development has been made possible by an ongoing extraordinary progress in process technology and design technology of integrated circuits made of silicon material. The progress in process technology is generally expressed in the increase in the number of transistors, the fundamental electronic devices, in one IC over the years. Over the past decades, an exponential growth of the number of transistors per IC could be observed at a rate that has been reported as a doubling of the number of transistors either every 12, 18 or 24 months. This exponential growth is called Moore's Law after a forecast of the Intel co-founder [85]. This forecast has become a goal of the semiconductor industry in the meantime.

Process technology contributes to achieving this goal by developing the ability to realize ever smaller transistor dimensions and the ability to manufacture ICs

- with ever increasing number of transistors

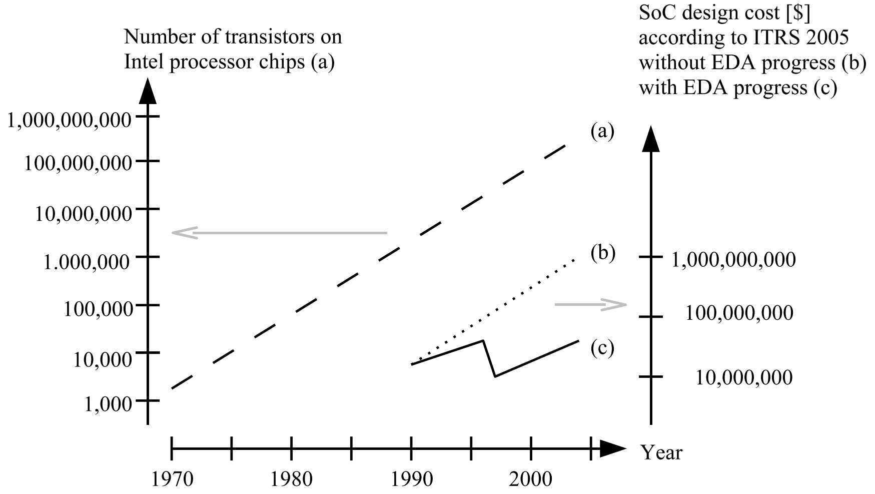


Figure 1. Trends in process and design technology. Data have been taken from the Intel web site and from the International Technology Roadmap for Semiconductors (ITRS) web site.

- at nearly constant percentage of ICs that pass the production test (yield) and
- with a reasonable life time (reliability).

Design technology contributes to achieving the exponential growth of IC complexity by developing the ability to design ICs with ever increasing number of functions by

- bottom-up library development methods for complex components and
- top-down system design methodologies
- for quickly emerging system classes.

As a result it has been possible to produce ICs with an exponentially growing number of transistors and functions while keeping the increase in the cost of design, manufacturing and test negligible or moderate. Figure 1 illustrates the technology and design development. The dashed line shows the increase in number of transistors on Intel processors, which exhibits a doubling of this number every 24 months for 35 years. Similar developments have been observed for other circuit classes like memory ICs or signal processors. The solid line shows the development of design cost for system-on-chip ICs. If no progress in design technology and electronic design automation (EDA) had been achieved since

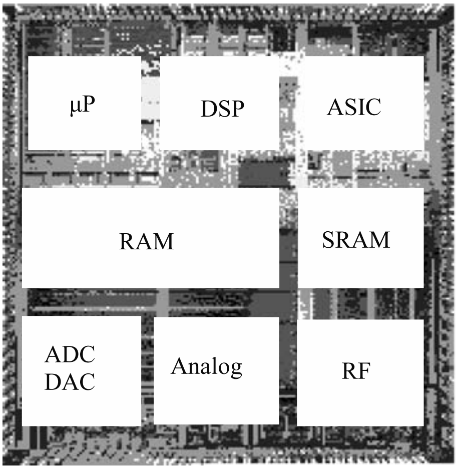


Figure 2. Mixed-signal system-on-chip ICs.

1990, design cost would have exhibited an exponential growth as indicated by the dotted line in Figure 1. But thanks to EDA innovations, design cost grew only slowly or even decreased. The decrease in the mid 90s for instance has been ascribed to small block reuse methodologies.

Because of the obvious economic advantage of integrated circuits, more and more system functionality moves into ICs, which become ever more heterogeneous [70]. Modern system-on-chip ICs, as sketched in Figure 2, contain for instance

- memory components, processors, application-specific ICs (ASICs),
- analog and digital components, high-frequency (RF) components,
- hardware and software components (embedded systems),
- optical or mechanical components.

But as miniaturization of transistors approaches atomic layers and as system complexity approaches billions of transistors, the challenge of IC design, manufacturing and test at reasonable cost has grown critically today. In addition, relative variations in process technology and the impact of technological effects on system level are increasing and cannot be compensated by circuit design alone. The result are significant variations in system performance, as for instance processor operating frequency, that have to be coped with. Statistical methods, which have been a topic of research and application in analog

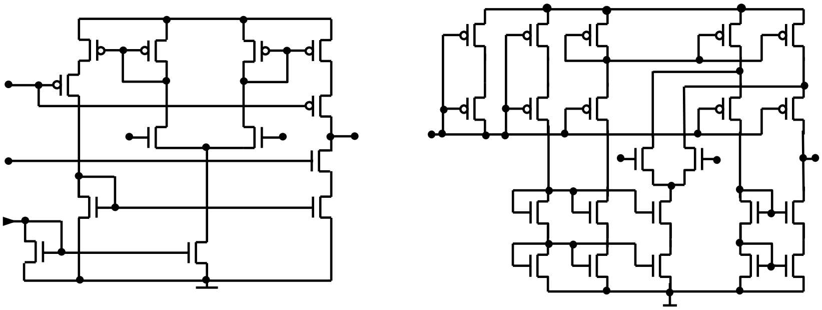


Figure 3. A CMOS operational transconductance amplifier (OTA, left) and a CMOS folded-cascode operational amplifier (right).

circuit design for decades, therefore become more and more important for digital design as well.

1.2 Analog Circuits

Analog components play an important role in integrated circuits. On one hand, important system functions like clock generation, or signal conversion between the off-chip analog environment and the on-chip digital signal processing, are realized by analog circuits. On the other hand, analog circuits are difficult to design and reluctant to design automation.

According to EDACafé Weekly (http://www.edacafe.com) on March 21, 2005, analog components use on average 20% of the IC area, as indicated in Figure 2. At the same time, the analog components require around 40% of the IC design effort and are responsible for about 50% of the design re-spins. Analog design automation is therefore urgently needed to improve the design quality and reduce the design effort, all the more as analog circuits are included in more and more ICs. EDACafé Weekly reported that 75% of all ICs would contain analog components by 2006.

Analog components appear in integrated circuits in various complexities. For a long time, analog design has been addressing analog integrated circuits as operational amplifiers (Figure 3) or filters (Figure 4). These circuits range from a few transistors to around hundred(s) of transistors and are designed based on numerical simulation with SPICE [89] and its advanced successors for the computer-aided evaluation of a circuit's performance.

But analog circuits on modern ICs are more complex and SPICE-like simulation consumes too much time so that more efficient modeling levels have to be entered. Examples for such complex analog circuits are analog/digital

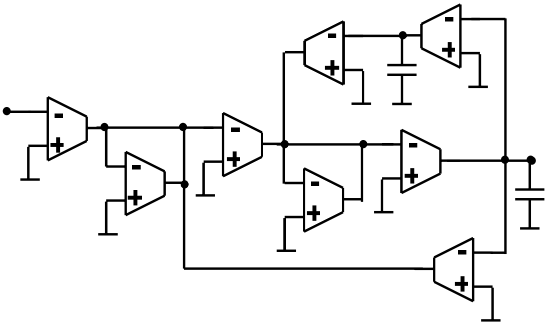


Figure 4. An OTA-C biquad filter with OTAs from Figure 3 as building blocks [57].

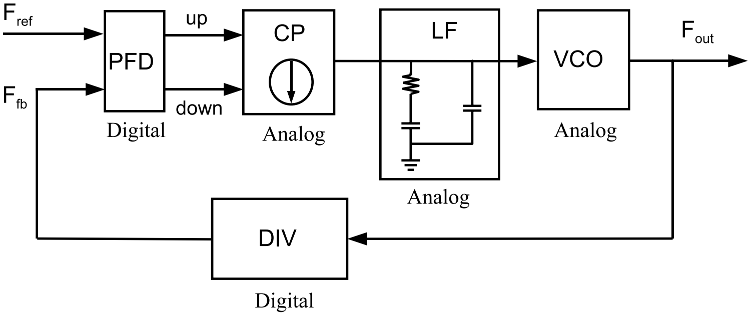


Figure 5. A phase-locked loop (PLL) with digital components, phase frequency detector (PFD), divider (DIV), and analog components, charge pump (CP), loop filter (LF), voltage-controlled oscillator (VCO).

converters (ADCs), digital/analog converters (DACs), or phase-locked loops (PLLs). These circuits are mixed-signal circuits with both analog and digital parts with an emphasis on the analog circuit behavior. Popular functions of PLLs are clock generation, clock recovery or synchronization. Figure 5 shows the structure of a charge-pump PLL that can be used to generate a high-frequency signal through an oscillator that is synchronized to a high-precision low-frequency signal in a control loop.

The complexity of these analog and mixed-signal circuits ranges around thousand(s) of transistors. The filter in Figure 4, which is hierarchically built from OTA building blocks, can easily be simulated flat on circuit level using the OTA transistor netlist in Figure 3, whereas the simulation time of the PLL

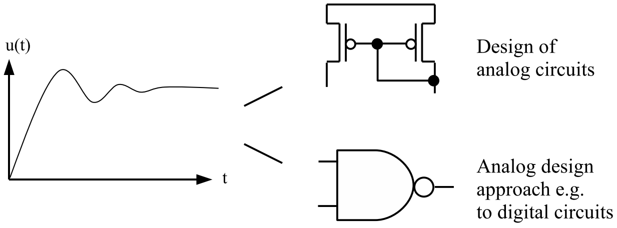

u(t)
t
Design of analog circuits
Analog design approach e.g. to digital circuits

Figure 6. Scope of analog design.

in Figure 5 flat on circuit level reaches hours or days. The usual design process that requires frequent numerical simulations becomes infeasible in this case. A remedy offers the transition from modeling on circuit level to modeling on architecture level using behavioral models. By modeling the whole PLL on architecture level or by combining behavioral models of some blocks with transistor models of other blocks, the PLL simulation time reduces to minutes and the usual design process remains feasible.

1.3 Analog Design

Analog design usually stands for the design of analog circuits. Its most important characteristic is the consideration of signals that are continuous in time and value. Contrary to that, digital design considers values that are discrete in time and value. Please note that value-continuous and time-continuous signals can be considered for any kind of circuit class, which leads to an extended meaning of analog design in the sense of an analog design approach (Figure 6).

Analog design therefore not only refers to the design of analog and mixed-signal circuits but includes for example the design of micromechanical components, the characterization of digital library components on circuit level, or the characterization of physical effects from interconnects or substrate.

As in digital design, the analog design process leads from a description of the system requirements to a complete hardware and software realization (i.e. implementation) in a cascade of individual design steps. Each design step can be classified in terms of the design levels, the design partitioning, and the design views it involves (Table 1).

The design partitioning and the design levels reflect the complexity of an integrated circuit, which is recursively decomposed into a hierarchy of sub-blocks that are designed individually and then composed to the system in a divide-and-conquer manner.

Table 1. Analog design terms.

Design Partitioning	Design level	Design view
HW/SW	System	Behavior
Analog/digital	Architecture	Structure
Functions, blocks	Circuit	Geometry
	Device, process	

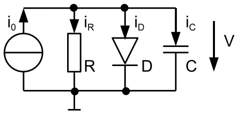
Typical analog design levels are system level, architecture level, circuit level, device level and process level. An example of an analog system is a receiver frontend, whose architecture consists of different types of blocks like mixers, low-noise amplifiers (LNAs) and ADCs. On circuit level, design is typically concerned with a transistor netlist and applies compact models of components like transistors. These compact transistor models are determined on device level requiring dopant profiles, which in turn are determined on process level, where the integrated manufacturing process is simulated.

The design partitioning concerns the hierarchical decomposition of the system into subblocks. Design partitioning is required to handle the design complexity and refers to important design decisions with respect to the partitioning between hardware and software, between analog and digital parts, and between different functions and blocks. The design partitioning yields graphs of design tasks that represent design aspects like block interdependencies or like critical paths for the design project management.

During the design process, a refinement of the system and an enrichment with more and more details concerning the realization occurs. Along with it, the system consideration traverses the design levels from the system level over the architecture level down to the circuit level. Moreover, the design process switches among three different design views, which are the behavioral view, the structural view and the geometric view (Table 2). The design level rather refers to the position of the considered system component within the hierarchical system (de)composition. The design view on the other hand rather refers to the modeling method used to describe the considered system component. The behavioral view refers to a modeling with differential equations, hardware and behavior description languages, transfer functions, or signal flow graphs. It is oriented towards a modeling of system components with less realization details. The structural view refers to a modeling with transistor netlists or architecture schematics. It is oriented towards a modeling of system components with more realization details. The geometric view refers to the layout modeling. On the lowest design level, it represents the ultimate realization details for IC production, the so-called mask plans. During the design process, not only a

Table 2. Some modeling approaches for analog circuits in behavioral and structural view.

Behavior	Linear/nonlinear differential equations
	Transfer function (e.g. pole/zero form)
	Behavior description code (e.g. VHDL, Verilog)
	Signal flow graph
Structure	Circuit netlist (R, L, C, transistor, source elements)
	LTI netlist (adding, scaling, integrating elements)
	Architecture schematic



$$\frac{dv}{dt} + \frac{1}{RC}v + \frac{I_S}{C}(\exp(\frac{v}{V_T}) - 1) - \frac{1}{C}i_0 = 0$$

Figure 7. Nonlinear analog circuit. Structural model in form of a circuit netlist, behavioral model in form of a differential equation. Concerning the output voltage, both models are equivalent.

refinement towards lower design levels takes place, but also a trend from the behavioral view towards the structural and geometric view.

In analog design, often the structural view is to the fore, as transistor netlists on circuit level or block netlists are prevalent in the design. But the behavioral view is immediately included, for instance by the implicit algebraic differential equations behind the transistor netlist or by program code. Behavioral and structural models can be equivalent as in the two examples in Figures 7 and 8 but they do not have to be equivalent.

The behavioral model and the structural model in Figure 8 represent more abstract models of the OTA circuit in Figure 3. While Figure 3 represents the OTA on circuit level using detailed transistor models, Figure 8 represents the OTA on architecture level to be used as a component of the filter in Figure 4. The architecture-level structural and behavioral models are less complex, less accurate and faster to simulate than the circuit-level models.

The relationship between design levels and design view in the design process has been illustrated by the so-called Y-chart, in which each branch corresponds to one of the three design views and in which the design levels correspond to

```
ENTITY ota is
      GENERIC (gdm:real;Cin:real;CT:real;Cout:real;Rout:real;);
      PORT (terminal Tplus, Tminus, Tout, Tvdd, Tvss ,
Tibias:electrical);
END ENTITY ota;
-- Simple Ota architecture
architecture behavioural_simple of ota is
---***external/internal quantities***
begin  -- behavioural_simple
---***input stage***
      icd == cin*vd'dot;
      vd == ird*rin;
---***transfer stage***
      iT == vd / rT;
      iT == icT + irT;
      vT == irT*rT;
      icT == cT*vT'dot;
---***output stage***
      vout == irout*rout;
      icout == cout*vout'dot;
      iout0 == vT *gdm;
      iout == iout0+icout+irout;
end behavioural_simple;
```

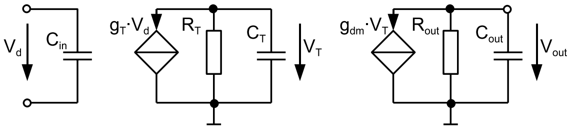


Figure 8. Simplified models of the OTA in Figure 3. Behavioral model in form of a description language code, Structural model in form of a circuit netlist. Concerning the OTA ports, both models are equivalent and approximate to the transistor netlist in Figure 3.

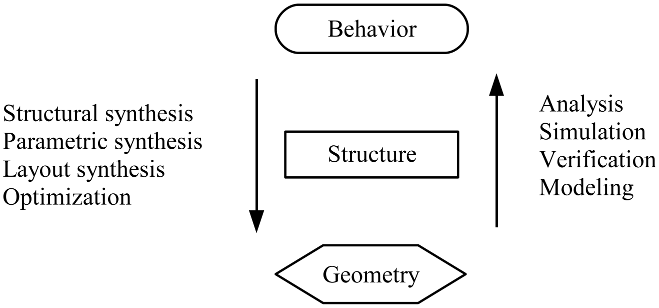


Figure 9. Analog synthesis and analysis.

concentric circles around the center of the Y [47]. From the design automation point of view, the change in design view during the design process from the behavioral view towards the structural and geometric view is of special interest. The term synthesis as the generic term for an automatic design process has therefore been assigned to the transition from the behavioral view towards the structural view, and from the structural view to the geometric view, rather than to the transitions in design level or design partitioning [84]. Figure 9 illustrates this concept, where synthesis and optimization denote the transition from behavioral to structural and geometric view, and where analysis and simulation denote the inverse transition.

In analog design, we can distinguish three synthesis phases regarding the transition from the behavioral towards the structural and geometric view. Starting from a behavioral formulation of the requirements on the performance of a system or system component, structural synthesis serves to find a suitable structure of the system component.

In the subsequent step, specific for analog design, parameter values for the given structure, as for instance transistor widths and lengths in CMOS technology, have to be determined by parametric synthesis. This process, and likewise the result, is also called analog sizing. Parametric synthesis can be interpreted as an additional part of the structural synthesis, which completes the structure with required attributes, i.e. sizing. After that, layout synthesis leads to the placement and routing.

This top-down process of synthesis and optimization [55, 24] is always accompanied by bottom-up steps of analysis and simulation. These are required for function modeling and evaluation during the synthesis and optimization process, or for verification after synthesis.

An exemplary design flow of a PLL is depicted in Figure 10 [128]. The design process starts on architecture level from a specification of the PLL behavior concerning the locking behavior, noise, power, stability, and output frequency.

Structural synthesis on architecture level (de)composes the PLL (into)of a netlist of five blocks. In this step, the design view changes from the behavioral view to a mixed structural-behavioral view: a netlist is obtained, whose components are modeled behaviorally. Additionally, an important partitioning into digital and analog blocks is carried out. The open boxes for the three analog blocks charge pump, loop filter and current-controlled oscillator indicate that the structure has not been completed yet. This is done in the next step, which is a parametric synthesis step on architecture level, where architectural parameter values of the three analog blocks are computed. For instance values for the charge current and jitter of the charge pump, values for the R's and C's of a passive loop filter, and values for the gain, current and jitter of the controlled oscillator are determined. Only now, the architectural realization of the PLL is complete.

The parametric synthesis step is done on architecture level using behavioral models for the PLL blocks for two reasons. First, the circuit-level transistor netlist of the PLL is much too expensive to simulate as mentioned before, which makes the parametric synthesis of a PLL on circuit level infeasible. Second, using behavioral models for the PLL blocks allows to hide more realization details at this stage of the design process and leads to a top-down specification propagation. However, there must be a bottom-up modeling of the performance capabilities of PLL blocks to prevent the parametric synthesis on architecture level from producing unrealistic requirements on the PLL blocks.

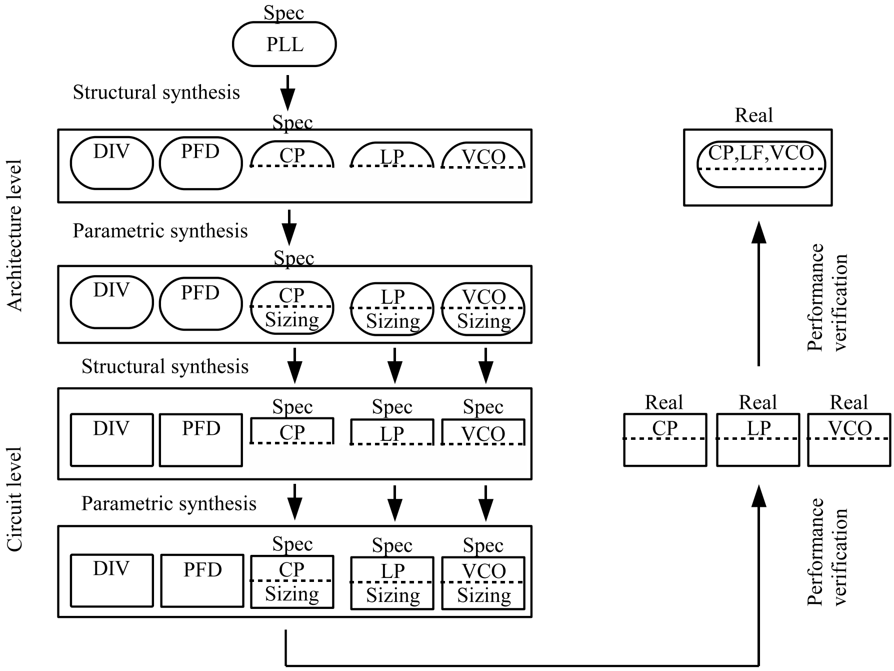


Figure 10. Design flow of a PLL. As in Figure 9, boxes denote the structural view, rounded boxes the behavioral view.

Next, the design level switches to the circuit level, where the same process from the architecture level is repeated for all blocks simultaneously, as indicated by three parallel arrows in Figure 10. A transistor netlist capable of satisfying the block specification is determined by structural synthesis. The circuit-level parameter values, as the transistor widths and lengths, are computed by parametric synthesis.

The transition from the architecture level to the circuit level includes a transition from the behavioral view of the three analog blocks “under design” to the structural view. Here a transformation of the architectural parameter values to a circuit-level performance specification of each individual block is required. Adequate behavioral block models do not only consider the input and output behavior but prepare the required transformation by including model parameters that have a physical meaning as performances of a block on circuit level, as for instance the gain of the VCO.

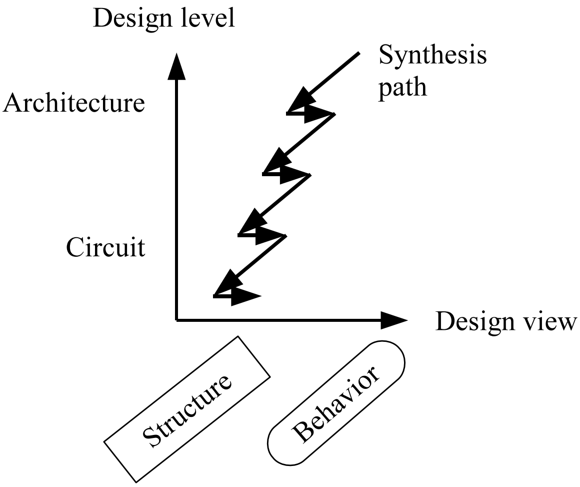


Figure 11. Analog synthesis path of a PLL.

After the synthesis process, the obtained real block performance values on circuit level can be used to verify the PLL performance.

If we illustrate the synthesis process just described in a plane with design level and design view as coordinates, we obtain a synthesis path as shown in Figure 11.

This figure reflects that both structural and parametric synthesis generally rely on function evaluations, which happen in a behavioral view. This leads to periodical switches to the behavioral view within the synthesis flow that can be illustrated with a “fir”-like path.

A top-down design process as just described requires on each level adequate models from the lower design levels. These models are computed beforehand or during the design process. Ideally, the results of the top-down process can be verified. But as the underlying models may not be accurate enough or incomplete, for instance for reasons of computational cost, verification may result in repeated design loops.

The design of the PLL was illustrated starting on the architecture level. On system level, signal flow graphs, which represent a yet non-electric information flow, are often applied. Switching to the architecture level then involves the important and difficult transition to electrical signals.

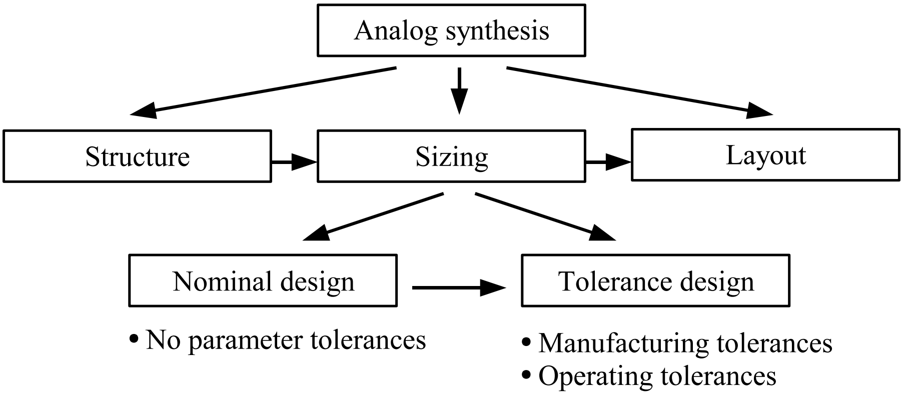


Figure 12. Analog sizing tasks.

1.4 Analog Sizing

Among the three synthesis steps concerning structure, sizing and layout, this book deals with the sizing step, regardless of the design level or hierarchical design stage in which it happens. Sizing is done in two steps (Figure 12).

In a first nominal design step, no parameter tolerances are considered, and sizing means optimizing nominal performance values subject to constraints that have to be satisfied.

In the subsequent tolerance design step, the inevitable variations of the manufacturing process and of the operating conditions are considered.

The variations of the manufacturing process are modeled by global variations of parameters like oxide thickness and by local variations of parameters like threshold voltage. Global parameter variations are usually modeled as equal for all transistors in a circuit, whereas local parameter variations are usually modeled as independent for each transistor in a circuit.

The variations of the operating conditions are specified by intervals of parameters like supply voltage or temperature for which has to be guaranteed that the system works properly.

The tolerance design task can be subdivided into subtasks as illustrated in Figure 13.

Worst-case optimization denotes the task of minimizing the worst-case performance deviation in a given tolerance region of parameters. This goal includes a minimum performance sensitivity with regard to parameter tolerances. Every optimization problem requires the evaluation of its optimization objective.

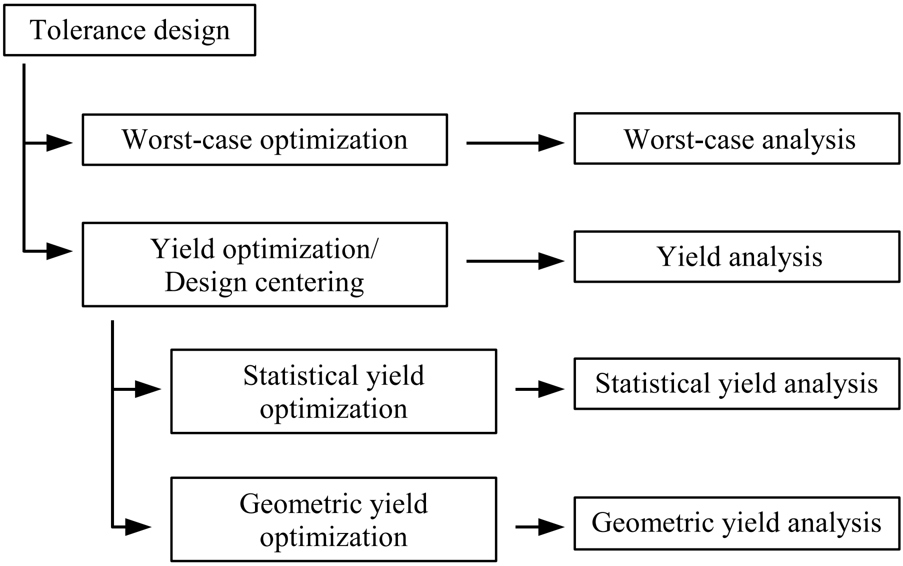


Figure 13. Tolerance design tasks.

Worst-case optimization requires the evaluation of worst-case performance values. This task in turn is denoted as worst-case analysis.

Yield optimization and design centering equally denote the task of maximizing the so-called yield, which is the percentage of systems that satisfy the performance specification despite performance variations due to manufacturing tolerances and operating tolerances.

There are two approaches to yield optimization/design centering, a statistical approach and a geometric approach.

Statistical-yield optimization is based on a statistical estimation of the yield. The underlying statistical yield analysis is done by means of a Monte-Carlo analysis. Please note that the term statistical-yield optimization refers to the statistical estimation of the yield, but not to the optimization method.

Geometric-yield optimization replaces the statistical estimation of yield by a yield approximation that is based on geometric considerations. The manufacturing variations are described by a probability density function of statistical parameters. Those parameter vectors that lead to a violation of the performance specification are cut away. A geometric yield analysis is now based on an approximation of the geometry of the performance specification in the statistical parameter space.

Table 3. Characteristics of numerical optimization.

Objective dimension:	Single-objective optimization	Multiple-objective optimization
Objective nature:	Deterministic objective	Statistical objective
Solution nature:	Deterministic optimization	Statistical optimization
Search basis:	Gradient-free approach	Gradient/sensitivity-based approach

The term design centering for yield optimization emanates from such a spatial interpretation of the optimization problem. Section 7.1.2 will show that a maximum yield is obtained when the center of gravity of the original probability density function hits the center of gravity of the truncated probability density function. Yield optimization as finding an equilibrium of the center of gravity of the manufactured "mass" concerning the performance specifications becomes design centering in this interpretation. This interpretation suggests that the terms yield optimization and design centering should be used as synonyms.

As indicated in Figure 13, we emphasize the distinction between the analysis of an objective and the optimization of this objective. We will see that yield optimization approaches result immediately from the two ways of either statistically estimating or geometrically approximating the yield. For that reason, the terms "statistical" and "geometric" will be used to distinguish yield optimization/design centering approaches. We will derive the first- and second-order gradient of the yield based on the results of a Monte-Carlo analysis in Section 7.1. This leads to the formulation of a deterministic optimization method for the statistically analyzed yield, which could be denoted as deterministic statistical-yield optimization approach. We will derive the gradients of the worst-case distances, which are geometric measures of yield, in Section 7.2.1, which leads to a deterministic geometric-yield optimization approach.

Another important issue about yield optimization/design centering is the difficulty to maximize yield values beyond 99.9%, which are described by terms like six-sigma design. We will explain how to deal with these cases in statistical-yield optimization in Section 4.8.7. Geometric-yield optimization inherently covers these cases, as illustrated in Section 6.3.9.

Analog sizing is a systematic, iterative process that applies mathematical methods of numerical optimization. The solution methods can be characterized in different ways, which are summarized in Table 3.

One way of characterization is to distinguish between single-objective and multiple-objective optimization problems. Nominal design with many performance features usually is amultiple-objective optimization problem. Statistical-yield optimization with the single objective yield on the other hand is a single-objective optimization problem. We will describe the two types of optimization problems in Sections 4.5 and 4.6. We will formulate yield optimization/design centering both as a single-objective and as a multiple-objective optimization approach in Section 4.8.5. This allows us to consider nominal design and tolerance design in a unifying manner.

Other ways of characterizing the optimization problems of nominal and tolerance design are according to the nature of the objectives or of the solution approaches. The objective and the solution approach either can be statistical or deterministic. Concerning yield optimization/design centering, we will introduce a way to closely couple the statistical objective yield with a deterministic objective worst-case distance in Sections 6.2.4, 5.5 and 6.3.7. This allows a unified view on the yield optimization/design centering tasks.

Another way to characterize optimization tasks is according to the usage of gradients in the solution approach. While statistical optimization methods often work without gradients, deterministic optimization methods often are based on gradients and second-order derivatives. In this work, deterministic solution methods for yield optimization/design centering that use gradients will be formulated in Chapter 7.

It is important to note that numerical optimization is not only required for yield optimization/design centering and performance optimization, but for tasks that are named analysis in Figure 13. Worst-case analysis as well as geometric yield analysis will turn out to be optimization problems as well. Solution methods for these tasks will be presented in Chapter 5 and Section 6.3.

Optimization methods applied to sizing generally have to frequently evaluate the optimization objective. In analog design this is done through numerical simulation with SPICE-like simulators or through a performance model that has to be computed a-priori based on numerical simulation. As a single numerical simulation is very expensive in terms of CPU time, and as many simulations are required within sizing, the number of required numerical simulations is the essential measure for the CPU cost of a sizing algorithm.

Table 4 illustrates nominal design and yield optimization/design centering in a typical sizing process of operational amplifiers like those shown in Figure 3. Five performance features (first column of Table 4) are considered in this case, which characterize the transfer function (gain, transit frequency), stability (phase margin), speed (slew rate) and power. For each performance feature, a lower or upper bound is specified that defines the range of full performance (second column).

Table 4. Sizing of an operational amplifier.

1 Performance feature	2 Specification feature	3 Initial	4 After nominal design	5 After yield optimization/design centering
Gain	$\geq 80dB$	$67dB$	$100dB$	$100dB$
Transit frequency	$\geq 10MHz$	$5MHz$	$20MHz$	$18MHz$
Phase margin	$\geq 60°$	$75°$	$68°$	$72°$
Slew rate	$\geq 10V/\mu s$	$4V/\mu s$	$12V/\mu s$	$12V/\mu s$
DC power	$\leq 50\mu W$	$122\mu W$	$38\mu W$	$39\mu W$
Yield		0%	89%	99.9%

Numerical simulation of an initial sizing of circuit parameters that could for instance originate from a previous process technology leads to the performance values in the third column. We can see that all performance features but the phase margin initially violate the respective performance-feature bounds.

A performance optimization performance optimization that aims at satisfying all performance-feature bounds with as much safety margin as possible results in the performance values given in the fourth column. All bounds are satisfied with a certain safety margin. We cannot decide if one performance safety margin is better than another one, but an analysis of the yield shows that 89% of all manufactured circuits will satisfy the performance specification after nominal design.

Next, yield optimization/design centering is executed and results in the performance values in the last column. Essentially, the safety margin for the transit frequency has been decreased while the safety margin for the phase margin has been increased. The effect is an increase of the yield to 99.9% after yield optimization/design centering.

The main reason for splitting the sizing into the two stages nominal design and tolerance design is the cost of numerical simulation. Including parameter tolerances and yield as an optimization objective leads to a significant increase in the required number of simulations during optimization. It is therefore advantageous to first optimize the circuits' safety margins as much as possible without inclusion of tolerances and yield.

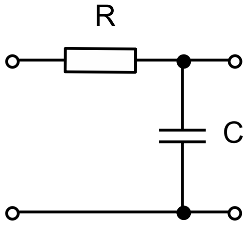
Chapter 2

TOLERANCE DESIGN: EXAMPLE

2.1 RC Circuit

Using an elementary RC circuit, we will illustrate the tasks of nominal design and tolerance design. Figure 14 shows the circuit netlist and circuit variables.

Two performance features, the time constant τ and the area A, and two circuit parameters, the resistor value R and the capacitor value C, are considered. Please note that R and C are normalized, dimensionless quantities. This example is well suited as we can explicitly calculate the circuit performance in dependence of the circuit parameters and as we can visualize the design situation in a two-dimensional performance space and a two-dimensional parameter space.



$$\begin{aligned} \tau &= R \cdot C \\ A &= R + C \end{aligned}$$

Figure 14. Elementary RC circuit with performance features time constant τ and area A as a function of resistor value R and capacitor value C.

2.2 Performance Evaluation

The right side of Figure 14 displays the performance functions of the RC circuit. The performance features, i.e. the time constant τ and the area A, for the given parameters, i.e. the resistor value R and the capacitor value C, can be explicitly evaluated in this case. But in general, performance evaluation requires expensive numerical simulation (Section 3.11).

Please note that the RC circuit is a linear circuit because it consists of linear circuit elements, as opposed to nonlinear circuit elements like transistors. Nevertheless the performance feature τ is nonlinear in the parameters.

2.3 Performance-Specification Features

For the RC circuit, a lower and upper bound on the time constant τ and an upper bound on the area A are specified:

$$\begin{array}{r} 0.5 \le \tau \le 2.0 \\ A \le 4.0 \end{array} \equiv \begin{array}{r} 0.5 \le\; R\cdot C \;\le 2.0 \\ R+C \le 4.0 \end{array} \tag{1}$$

We call the inequality resulting from one bound a performance-specification feature. (1) hence formulates three performance-specification features of the RC circuit.

2.4 Nominal Design/Performance Optimization

Nominal design naturally tries to keep the nominal performance within the specified limits with as much safety margin as possible. In our example, the time constant τ has to be centered between its lower and upper bound, and the safety margin of the area A with regard to its upper bound has to be maximized. This is achieved by inspection of the problem or through the optimality conditions of the following optimization problem:

$$\begin{array}{rl} & \min\ A \quad \text{subject to} \quad \tau = 1.25 \\ \equiv & \min\ R+C \ \text{ s.t. } \ R\cdot C = 1.25 \end{array} \tag{2}$$

The solution results in:

$$\begin{aligned} R' &= C' = \sqrt{5}/2 \approx 1.118 \\ \tau' &= 1.25 \\ A' &= \sqrt{5} \approx 2.236 \end{aligned} \tag{3}$$

Figure 15 illustrates the situation of the RC circuit after nominal design. Figure 15(a) sketches the two-dimensional space spanned by the performance features "area" and "time constant," Figure 15(b) shows the corresponding two-dimensional space spanned by the parameters R and C.

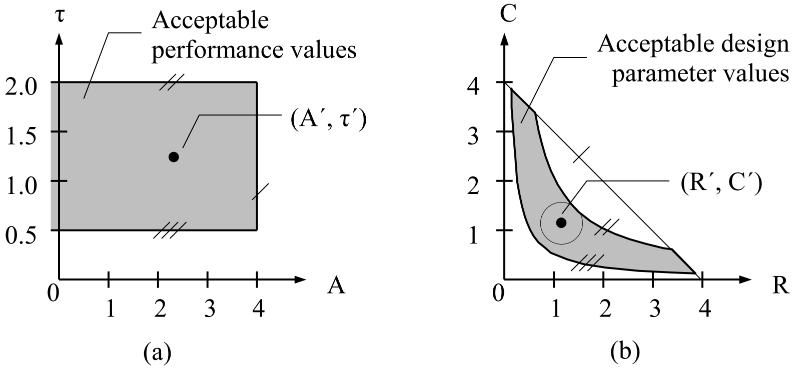


Figure 15. RC circuit. *(a)* Performance values of RC circuit after nominal design. *(b)* Parameter values of RC circuit after nominal design.

A performance specification as in (1) constitutes box constraints as illustrated by the gray box in Figure 15(a). In this example, the box of acceptable performance values is not closed as no lower bound for the area is given. Figure 15(a) shows the nominal performance values τ' and A' and their centering between the lower and upper bound of the time constant τ. From Figure 15(a) it is not yet clear why the safety margin of the area to its upper bound is not larger.

Figure 15(b) shows the corresponding situation for the parameters. Using the performance equations in Figure 14, the performance specification can be formulated as inequalities for the parameters as done in (1). This results in a bounding curve for each of the three performance-specification features in Figure 15(b). The correspondence between the bounding curves in Figure 15 is indicated by single, double and triple slashes. The single slash belongs to the performance-specification feature $A \leq 4.0 \equiv C = 4.0 - R$. The double slash belongs to the performance-specification feature $\tau \leq 2.0 \equiv C \leq 2.0/R$, and the triple slash belongs to $\tau \geq 0.5 \equiv C \geq 0.5/R$. We can see that the region of acceptable parameter values is closed. The nominal parameter values R' and C' are also shown. Here we can graphically reproduce the nominal design process. Centering the design between the lower and upper bound of the time constant is achieved by all points on the curve $C = 1.25/R$ that runs between the double-slashed curve and the triple-slashed curve. From all points on that curve, R' and C' is the one with the largest safety margin to the single-slashed line representing the upper bound on the area. A larger safety margin for the upper area bound could only be achieved at the cost of the safety margin for the lower bound on the time constant.

We can also see that although we have centered the performance values, we have not centered the parameter values: R' and C' are closer to the double-slashed curve than to the triple-slashed curve. This difference is due to the nonlinearity of the time constant in the parameters. If we had centered the parameter values instead of the performance values, using $R = C = x$, we would not have centered x^2 between 0.5 and 2, i.e. $x^2 = 0.5 \cdot (0.5 + 2)$, but we would have centered x between $\sqrt{0.5}$ and $\sqrt{2}$, i.e. $x = 0.5 \cdot (\sqrt{0.5} + \sqrt{2})$. As a result, we would have designed the following parameter and performance values:

$$\begin{aligned} R'' &= C'' = 0.75 \cdot \sqrt{2} \approx 1.061 \\ \tau'' &= 1.125 \\ A'' &= 1.5 \cdot \sqrt{2} \approx 2.121 \end{aligned} \tag{4}$$

This difference between the situation in the performance space and the parameter space leads us to yield optimization/design centering.

2.5 Yield Optimization/Design Centering

In real designs, we are additionally facing statistical variations in the parameters. As a result, parameters do not have deterministic values, but are random variables that are statistically distributed according to a certain statistical distribution. Let us assume that in our example the parameters R and C are normally distributed. Let the standard deviations of R and C be $\sigma_R = 0.2$ and $\sigma_C = 0.8$, and let the correlation between R and C be $\varrho = 0$. Then the joint normal probability density function $\text{pdf}(R, C)$ after nominal design is:

$$\text{pdf}(R, C) = \frac{1}{2\pi\sigma_R\sigma_C} \cdot \mathrm{e}^{-0.5\left(\left(\frac{R-R'}{\sigma_R}\right)^2 + \left(\frac{C-C'}{\sigma_C}\right)^2\right)} \tag{5}$$

A probability density function is the continuous counterpart to a relative frequency distribution. A relative frequency distribution describes the percentage of events in each class of parameter values among all events. For a probability density function this corresponds to the probability of occurrence in an infinitesimal parameter class $(R + dR) \cdot (C + dC)$, which is given by $\text{pdf}(R, C) \cdot (R + dR) \cdot (C + dC)$. Due to the quadratic form in the exponent of (5), the probability density function of a normal distribution looks like a bell-shaped curve. The level curves of our two-dimensional probability density function are ellipses around the nominal parameter values R', C'. At R', C' the probability density function takes its maximum value indicating maximum probability of occurrence. The "larger" a level ellipse is, the smaller is the corresponding constant value of the probability density function indicating smaller probability of occurrence.

Figure 16 illustrates the situation for the RC circuit after nominal design. To indicate the decreasing values of the probability density function, the level

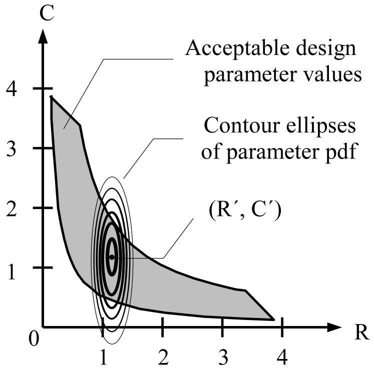

C
4
3
2
1
0
1
2
3
4
R
Acceptable design parameter values
Contour ellipses of parameter pdf
(R′, C′)

Figure 16. Parameter values and parameter tolerances of the RC circuit after nominal design.

ellipses have decreasing line thickness. Had the standard deviations of R and C been equal, the level curves would have been circles. σ_C being larger than σ_R corresponds to a higher dilation of the level curves in the direction of C than in the direction of R. The distribution of the resistor values hence is less broad than that of the capacitor values. Figure 16 shows that parameter vectors exist that lead to a violation of the performance specification. The volume under the probability density function over the whole parameter space is 1, which corresponds to 100% of the manufactured circuits. We can see that a certain percentage of circuits will violate the performance specification. The percentage of manufactured circuits that satisfy the performance specification is called parametric yield Y. Parametric variations refer to continuous variations within a die or between dies of the manufacturing process. The resulting performance variations can be within the acceptable limits or violate them. In addition, the manufacturing process is confronted with catastrophic variations that result for instance from spot defects on the wafer causing total malfunction and resulting in the so-called catastrophic yield loss. The yield is an important factor of the manufacturing cost: the smaller the yield, the higher the manufacturing cost. In the rest of the paper, the term yield will refer to the parametric yield of a circuit.

The estimated yield after nominal design in this example is:

$$Y' = Y(R', C') = 58.76\% \tag{6}$$

Chapter 6 will deal with methods to estimate the parametric yield. Figure 17 extends Figure 16 with a third axis for the value of the probability density function to a three-dimensional view.

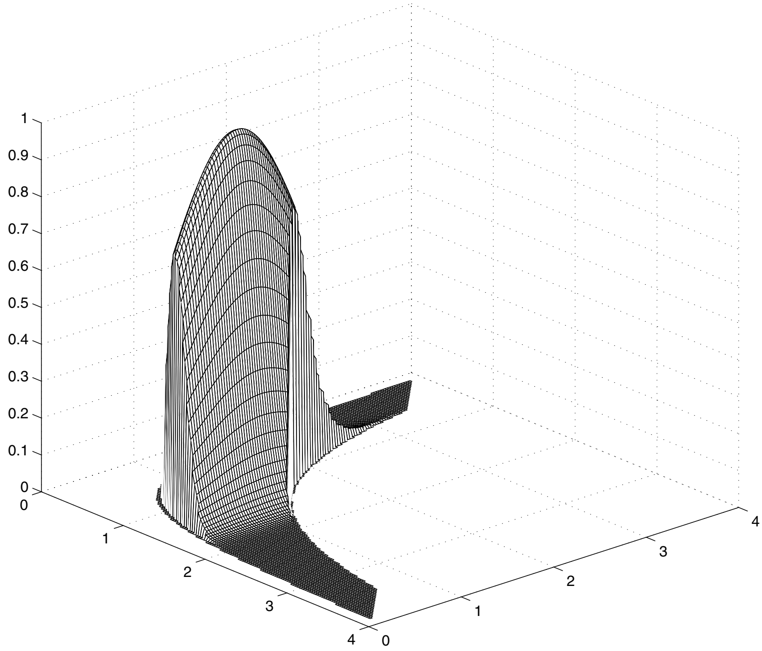


Figure 17. Probability density function after nominal design of the RC circuit, truncated by the performance specification.

Only that part of the probability density function that corresponds to acceptable values of R and C is given. We can think of the probability density function as a bell-shaped "cake" that describes the variation of the actual parameter vectors due to the manufacturing process. A portion of varying parameter vectors are cut away by the performance specification. An optimal design will lead to a maximum remaining volume of the "cake," which is equivalent to a maximum parametric yield Y_{max}. The part of the design process that targets at maximum yield is equally called yield maximization or design centering.

Figures 16 and 17 illustrate that the result of nominal design does not lead to a maximum yield. The first reason for that is the mentioned nonlinearity of the performance with respect to the parameters, which leads to skewing equal safety margins of the performance with respect to its lower and upper bounds in the parameter space. The second reason is the parameter distribution, which

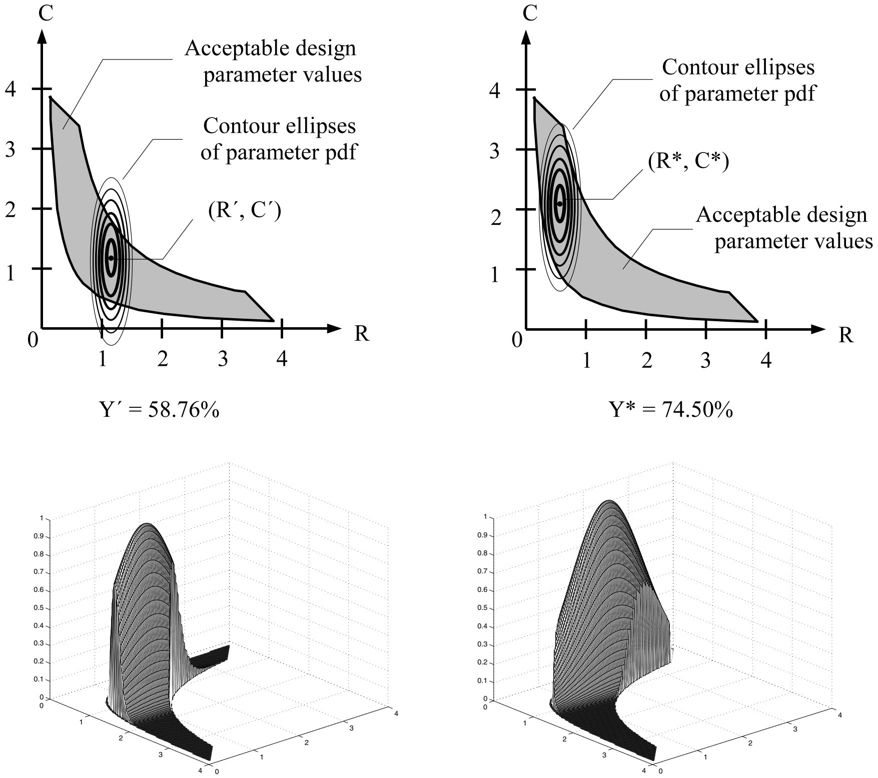


Figure 18. *(left)* Parameter values and truncated probability density function. after nominal design of the RC circuit. *(right)* Parameter values and truncated probability density function after yield optimization/design centering of the RC circuit.

leads to skewed spreading of the parameter values depending on the parameter correlation and variance values.

From Figures 16 and 17 it is intuitively clear that a simultaneous decrease of the nominal value of R and increase of the nominal value of C will increase the amount of volume of the "cake" that is not cut away by the performance specification and hence increase the yield. Figure 18 illustrates this. The increase in yield can be seen in the larger probability density function "cake" left by the performance specification and in the increased number of level ellipses that are completely inside the region of acceptable parameter values.

After yield optimization/design centering, the estimated yield in this example is:

$$Y^\star = Y(R^\star, C^\star) = Y(0.569, 2.016) = 74.50\% \tag{7}$$

Chapter 7 will deal with methods for yield maximization/design centering.

Please note that the complexity of performance evaluation through numerical simulation leads to a much more challenging yield optimization/design centering situation in real analog designs.

Chapter 3

PARAMETERS & TOLERANCES, PERFORMANCE & SPECIFICATION

3.1 Parameters

In the literature, the term parameter usually denotes a variable quantity that is related to the circuit structure and the circuit behavior. In this book, the term parameters shall be used in a more specific sense for those quantities that are input to the numerical simulation. Quantities related to the output of numerical simulation shall be denoted as performance features. This distinction is owing to the fact that the mapping of parameter values onto values of performance features by means of numerical simulation takes the – computationally very expensive – center stage of a systematic analog design process.

Definition: Parameters *are variable quantities of an analog system or an analog component that are input quantities to numerical simulation. The elements x_k of the parameter vector $\mathbf{x} = [x_1 \ldots x_k \ldots x_{n_x}]^T \in \mathbb{R}^{n_x}$ denote the values of the respective parameters, whose names are addressed through the index k.*

We can distinguish three types of parameters:

- Design parameters $\mathbf{x}_d = [x_{d,1} \ldots x_{d,k} \ldots x_{d,n_{xd}}]^T \in \mathbb{R}^{n_{xd}}$ are subject to the sizing process. Appropriate design parameter values have to be calculated such that the performance is “optimal”. In the RC circuit in Section 2.1, the capacitance and the resistance are design parameters. In CMOS circuits, mainly transistor geometries, i.e. channel widths and lengths, form the design parameters.
- Statistical parameters $\mathbf{x}_s = [x_{s,1} \ldots x_{s,k} \ldots x_{s,n_{xs}}]^T \in \mathbb{R}^{n_{xs}}$ are subject to statistical variations for instance due to fluctuations in the manufacturing process. The statistical variations are modeled in form of probability

density functions of the statistical parameters. In the RC circuit in Section 2.1, the capacitance and the resistance are statistical parameters. In CMOS circuits, mainly transistor model parameters, like for instance oxide thickness, threshold voltage or channel length reduction, form the statistical parameters.

In the RC circuit, design parameters and statistical parameters are identical. This corresponds to an analog circuit with discrete elements, as against to an analog integrated circuit. Discrete resistors for instance can be selected with different nominal values and with different tolerance classes, for instance 5%, 10%. The situation is different for integrated CMOS circuits. They have transistor widths whose values can be devised individually. However, the manufacturing process leads to a global variation of all transistors' widths, which is equal for all transistors and which is modeled in a statistical distribution of one parameter "width reduction." Another example of a statistical parameter is the oxide thickness. Width reduction and oxide thickness are not to be devised within analog design but within process design. In integrated circuits, we usually have separate sets of design parameters on the one hand and statistical parameters on the other hand. We will see in the course of this book that this discrimination makes statistical approaches to yield optimization/design centering at practicable computational cost very difficult.

- Range parameters $\mathbf{x}_r = [x_{r,1} \ldots x_{r,k} \ldots x_{r,n_{xr}}]^T \in \mathbb{R}^{n_{xr}}$ are parameters that are subject to a range of values they can acquire. Typical examples of range parameters are so-called operating parameters that model operating conditions, as for instance supply voltage or temperature. Range parameters differ from statistical parameters in that they do not have probability data in form of statistical distributions on top of their given ranges of values. For the temperature for instance, a range of -40°C to 125°C can be specified. Regardless of the distribution of the temperature, the performance has to be guaranteed for any temperature in this interval. The circuit operating conditions usually are a part of the circuit specification, but as the operating quantities are parameters, this part of the specification is categorized as parameter tolerances within the numerical-simulation-based sizing process.

3.2 Parameter Tolerances

We distinguish two types of parameter tolerances, parameter ranges and parameter distributions. Correspondingly, we have defined range parameters and statistical parameters. Typical examples of parameter tolerances and the essential sources of IC performance variation are the manufacturing fluctuations and the operating conditions. The inevitable fluctuations in the manufacturing process result in a whole range of performance values after production and test.

The manufactured ICs are arranged by quality classes, for instance of faster or slower ICs, and usually sold at different prices. The IC that is actually bought therefore can have a quite different performance from what is specified as its nominal performance in the data sheet. In addition, its performance will dynamically vary with altering operating conditions. These dynamic variations during operation may also be larger or smaller, depending on the quality of the actual IC. Operating conditions are usually modeled as intervals of range parameters. Manufacturing fluctuations are modeled as statistical parameter distributions on the regarded design level. The calculation of the statistical parameter distribution is a difficult task that requires specific optimization algorithms and the consistent combination of measurements that belong to different design levels and design views [40, 29, 109, 87, 98].

3.3 Range-Parameter Tolerances

Tolerances of range parameters are described by the region $T_r \subset \mathbb{R}^{n_{xr}}$ of values that the range parameters can adopt. The tolerance region T_r of range parameters can be defined in different ways. Typical tolerance regions are boxes, polytopes, ellipsoids or a general nonlinear region:

- box region

$$\begin{gathered} T_{r,B} = \{\mathbf{x}_r \mid \mathbf{x}_{r,L} \leq \mathbf{x}_r \leq \mathbf{x}_{r,U}\} \\ \mathbf{x}_{r,L}, \mathbf{x}_{r,U} \in \mathbb{R}^{n_{xr}} \ , \ \ -\infty \leq x_{r,L,i} < x_{r,U,i} \leq +\infty \end{gathered} \tag{8}$$

- polytope region

$$\begin{gathered} T_{r,P} = \{\mathbf{x}_r \mid \mathbf{A}_{poly} \cdot \mathbf{x}_r \leq \mathbf{b}_{poly}\} \\ \mathbf{A}_{poly} \in \mathbb{R}^{n_{poly} \times n_{xr}} \ , \ \mathbf{b}_{poly} \in \mathbb{R}^{n_{poly}} \end{gathered} \tag{9}$$

- ellipsoid region

$$\begin{gathered} T_{r,E} = \left\{\mathbf{x}_r \;\middle|\; (\mathbf{x}_r - \mathbf{x}_{r,0})^T \cdot \mathbf{A}_{ellips} \cdot (\mathbf{x}_r - \mathbf{x}_{r,0}) \leq b_{ellips}^2 \right\} \\ \mathbf{A}_{ellips} \in \mathbb{R}^{n_{xr} \times n_{xr}} \ , \ \mathbf{A}_{ellips} \text{ symmetric, positive definite} \end{gathered} \tag{10}$$

- nonlinear region

$$\begin{gathered} T_{r,N} = \{\mathbf{x}_r \mid \boldsymbol{\varphi}_{nonlin}(\mathbf{x}_r) \geq \mathbf{0}\} \\ \boldsymbol{\varphi}_{nonlin}, \mathbf{0} \in \mathbb{R}^{n_{nonlin}} \end{gathered} \tag{11}$$

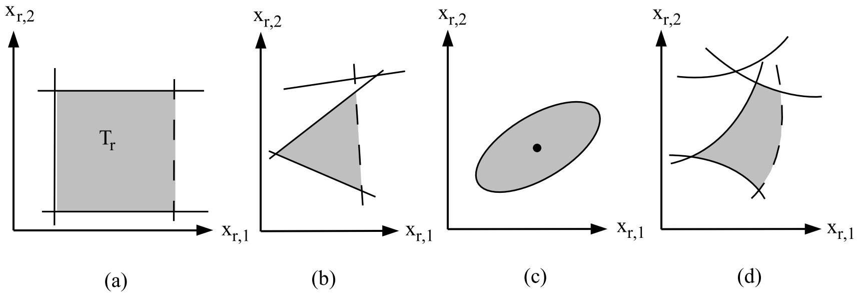


Figure 19. *(a)* Box tolerance region. *(b)* Polytope tolerance region. *(c)* Ellipsoid tolerance region. *(d)* Nonlinear tolerance region.

A vector inequality of the form $\mathbf{x} \leq \mathbf{y}$ assumes that $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{n_x}$ and is defined as

$$\mathbf{x} \leq \mathbf{y} \quad \Leftrightarrow \quad \underset{\mu}{\forall} \; x_\mu \leq y_\mu \tag{12}$$

Figure 19 illustrates the four types of tolerance regions. The box region according to (8), illustrated in Figure 19(a), is the prevalent type of range-parameter tolerances. It results if minimum and maximum values for each range parameter are specified. The dashed line in Figure 19(a) for instance corresponds to the upper bound of parameter $x_{r,1} \leq x_{r,U,1}$. In many cases, the tolerance region T_r will be closed, that means a finite range of values will be given for each range parameter. This reflects practical design situations where neither process parameters nor operating parameters should attain excessive values, be it for physical limitations or limited resources. The oxide thickness as well as the channel length for example are process quantities that cannot go below certain values. On the other hand, the channel length will be bounded above due to the limited chip area. For the operating range in turn, reasonable finite ranges have to be negotiated between manufacturer and customer. The size of this range is related to the product price: an IC that works between -40°C and 125°C will be more expensive than the same IC designed to work between 0°C and 70°C.

If the range parameters' bounds are determined in a linear dependence of each other, the tolerance region will have the shape of a polytope according to (9), as illustrated in Figure 19(b). (9) describes a set of n_{poly} inequalities of the form $\mathbf{a}_{\mu\cdot}^T \cdot \mathbf{x}_r \leq b_{poly,\mu}$, where $\mathbf{a}_{\mu\cdot}^T = [a_{\mu,1}\, a_{\mu,2}\, a_{\mu,3}\, \ldots]^T$ is the μ-th row of the matrix $\mathbf{A}_{poly}$. Every inequality corresponds to a bounding plane as indicated by the dashed line in Figure 19(b), determining a halfspace of acceptable parameter vectors. The tolerance region T_r is the intersection of all these halfspaces. Inequalities may be redundant, leaving them out would not alter the tolerance region T_r. The bounding line at the top of Figure 19(b) corresponds to such a redundant inequality.

(10) defines a tolerance region that is shaped as an ellipsoid, as illustrated by the ellipse in Figure 19(c). It will be detailed in the following Section 3.4 that the level curves of a Gaussian probability density function are ellipsoids. If the shape of a parameter region corresponds to the shape of the level contours of a probability density function, we are able to complement the parameter region with probability values for the occurrence. In this way, we allow a transition between range parameters and statistical parameters.

(11), illustrated by Figure 19(d), finally provides the most general description of a nonlinear tolerance region described by single nonlinear inequalities, $\varphi_{nonlin,\mu}(\mathbf{x}_r) \geq 0$, as nonlinear bounding surfaces, as indicated by the dashed line in Figure 19(d).

3.4 Statistical Parameter Distribution

The manufacturing variations are modeled through a joint, multidimensional, continuous distribution of statistical parameters.

In the discrete case, distributions are characterized by dividing the complete value range into an ordered sequence of intervals and counting the number of events in each interval. By normalizing these numbers to the total number of events, we obtain the relative frequency function. By accumulating the relative frequencies of all parameter intervals up to a maximal parameter value, we obtain the cumulative frequency function.

In the continuous case, the relative frequency function becomes a probability density function pdf, and the cumulative frequency function becomes a cumulative distribution function cdf. pdf and cdf are obtained from each other in the following way:

$$\mathrm{cdf}(\mathbf{x}_s) = \int_{-\infty}^{x_{s,1}} \ldots \int_{-\infty}^{x_{s,n_{xs}}} \mathrm{pdf}(\mathbf{t}) \cdot d\mathbf{t}\,, \quad d\mathbf{t} = dt_1 \cdot dt_2 \cdot \ldots \cdot dt_{n_{xs}} \tag{13}$$

$$\mathrm{pdf}(\mathbf{x}_s) = \frac{\partial^{n_{xs}}}{\partial x_{s,1} \cdot \partial x_{s,2} \cdot \ldots \cdot \partial x_{s,n_{xs}}} \, \mathrm{cdf}(\mathbf{x}_s) \tag{14}$$

In (13), the accumulation of relative frequencies has become an integration over the probability density function. The probability density function vice versa results from the partial derivatives of the cumulative distribution function with respect to all parameters.

For an interpretation of the cumulative distribution function values as relative portions of occurrence, we require a normalization of the probability density function such that:

$$\lim_{\mathbf{x}_s \rightarrow \infty} \mathrm{cdf}(\mathbf{x}_s) = \int_{-\infty}^{+\infty} \ldots \int_{-\infty}^{+\infty} \mathrm{pdf}(\mathbf{t}) \cdot d\mathbf{t} = 1 \tag{15}$$

An exemplary continuous distribution is the normal distribution, which is also called Gaussian distribution.

3.5 Univariate Normal Distribution

The cumulative distribution function and the probability density function of a univariate normal distribution are:

$$\mathrm{pdf}_N(x_s) = \frac{1}{\sqrt{2\pi} \cdot \sigma}\, \mathrm{e}^{-\frac{1}{2}\left(\frac{x_s - x_{s,0}}{\sigma}\right)^2} \quad , \quad \sigma > 0 \tag{16}$$

$$\mathrm{cdf}_N(x_s) = \int_{-\infty}^{x_s} \mathrm{pdf}_N(t) \cdot dt = \int_{-\infty}^{\frac{x_s - x_{s,0}}{\sigma}} \underbrace{\frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-\frac{1}{2}t^2}}_{\mathrm{pdf}_{NN}(t)} \cdot dt \tag{17}$$

$$= \frac{1}{2} + \int_0^{\frac{x_s - x_{s,0}}{\sigma}} \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt = \frac{1}{2} + \phi_0\left(\frac{x_s - x_{s,0}}{\sigma}\right) \tag{18}$$

$$= \frac{1}{2} + \frac{1}{2}\, \frac{2}{\sqrt{\pi}} \int_0^{\frac{x_s - x_{s,0}}{\sqrt{2}\cdot\sigma}} \mathrm{e}^{-t^2} \cdot dt = \frac{1}{2} + \frac{1}{2}\, \mathrm{erf}\left(\frac{x_s - x_{s,0}}{\sqrt{2} \cdot \sigma}\right) \tag{19}$$

In (16), $x_{s,0}$ denotes the mean value (expected value, expectation value) of parameter x_s and σ denotes the standard deviation of x_s. Background material about expectation values can be found in Appendix A. A random variable[1] x_s that originates from a normal distribution with mean value $x_{s,0}$ and standard deviation σ is also denoted as:

$$x_s \sim \mathcal{N}(x_{s,0}, \sigma^2) \tag{20}$$

[1] Speaking of a random variable x, we immediately denote the VALUE x of a random variable X. This value x refers e.g. to

- a random number generated according to the statistical distribution of the random variable,
- the argument in the probability density $\mathrm{pdf}(x)$,
- the infinitesimal range $x + dx$ having a probability of occurrence $\mathrm{prob}(X \leq x) = \int_x^{x+dx} \mathrm{pdf}(t)dt = \mathrm{cdf}(x + dx) - \mathrm{cdf}(x)$.

The probability that a random variable X has a value less or equal x is $\mathrm{prob}(X \leq x) = \int_{-\infty}^{x} \mathrm{pdf}(t)dt = \mathrm{cdf}(x)$. Please note that the probability that a random variable X has a value of x is $\mathrm{prob}(X = x) = \int_x^x \mathrm{pdf}(t)dt = 0$!

Table 5. Selected cumulative distribution function values of the univariate normal distribution.

$x_s - x_{s,0}$	-3σ	-2σ	$-\sigma$	0	$+\sigma$	$+2\sigma$	$+3\sigma$	$+4\sigma$
$\text{cdf}(x_s - x_{s,0})$	0.1%	2.2%	15.8%	50%	84.1%	97.7%	99.8%	99.99%

σ^2 denotes the variance of the univariate normal distribution.

In (17), the cumulative distribution function is formulated as an integral over the probability density function according to (13). By variable substitution, the integrand is simplified to the standard normal distribution with the mean value 0 and the standard deviation 1:

$$\frac{x_s - x_{s,0}}{\sigma} \sim \mathcal{N}(0, 1) \tag{21}$$

The standard normal probability density function pdf_{NN} in (17) can also be obtained by setting $x_{s,0} = 0$ and $\sigma = 1$ in (16). The cumulative distribution function of the variable x_s taken from a normal distribution is hence obtained by the standard cumulative distribution function value at $\frac{x_s - x_{s,0}}{\sigma}$, which denotes the difference of x_s to the mean value as a multiple of the standard deviation.

In (18) and (19), the cumulative distribution function is formulated using the function ϕ_0 and the error function erf, which are available in statistical tables and program libraries.

Figure 20 illustrates the probability density function pdf_N and the cumulative distribution function cdf_N of a univariate normal distribution.

pdf_N shows the typical bell shape of the normal distribution. As pdf_N is symmetrical around $x_{s,0}$, $\text{cdf}_N(x_{s,0}) = 0.5$. The hatched area under pdf_N from $-\infty$ till $x_{s,0} + \sigma$ is determined by $\text{cdf}_N(x_{s,0} + \sigma)$ according to (13), as indicated by the curved arrow in Figure 20. Vice versa, according to (14), the slope at $\text{cdf}_N(x_{s,0} + \sigma)$ is determined by $\text{pdf}_N(x_{s,0} + \sigma)$.

Table 5 shows selected cumulative distribution function values of the univariate normal distribution. It can be seen that there is a unique relation between tolerance intervals and yield values in the univariate case:

$$\begin{aligned} Y &= \text{prob}\{x_s \in T_s\} = \text{prob}\{x_s \in [x_{s,L}\ ,\ x_{s,U}]\} \\ &= \text{cdf}(x_{s,U}) - \text{cdf}(x_{s,L}) \end{aligned} \tag{22}$$

In this case, yield optimization/design centering can equivalently be formulated as centering the nominal parameter value between the interval bounds or as maximizing the interval bounds around the nominal parameter value.

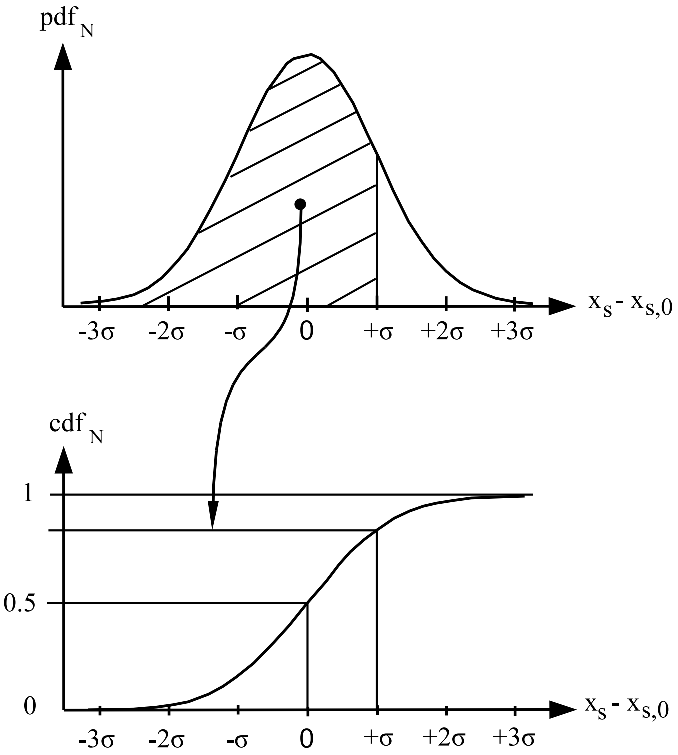


Figure 20. Probability density function pdf and cumulative distribution function cdf of a univariate normal distribution.

The relationship between tolerance ranges and yield values becomes more complicated for multivariate distributions.

3.6 Multivariate Normal Distribution

The probability density function of a multivariate normal distribution is given by:

$$\mathrm{pdf}_N(\mathbf{x}_s) = \frac{1}{\sqrt{2\pi}^{\,n_{xs}} \cdot \sqrt{\det \mathbf{C}}} \cdot \exp\left(-0.5 \cdot \beta^2(\mathbf{x}_s)\right) \quad (23)$$

$$\beta^2(\mathbf{x}_s) = (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \quad (24)$$

$$\beta \geq 0$$

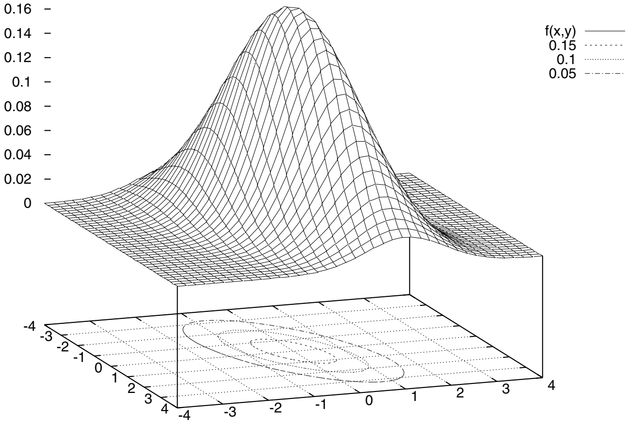


Figure 21. Multivariate normal distribution for two parameters according to (23), (24) or (34), respectively, with mean values $x_{s,0,1} = x_{s,0,2} = 0$, correlation $\varrho = 0.5$, and variances $\sigma_1 = 2$, $\sigma_2 = 1$.

In (24), the vector $\mathbf{x}_{s,0} = [x_{s,0,1} \ldots x_{s,0,k} \ldots x_{s,0,n_x}]^T$ denotes the vector of mean values $x_{s,0,k}\,,\ i = 1, \ldots, n_{xs}$, of the multivariate normal distribution of the parameters $x_{s,k}\,,\ k = 1, \ldots, n_{xs}$. Background material about expectation values can be found in Appendix A. Figure 21 illustrates the probability density function of a multivariate normal distribution for two parameters.

It shows the typical bell shape of the probability density function of two parameters and the quadratic form of the equidensity contours determined by (24).

The matrix **C** denotes the variance/covariance matrix, or covariance matrix, of the parameter vector $\mathbf{x}_s$.

A random vector $\mathbf{x}_s$ (see Note 1 in Section 3.5) that originates from a normal distribution with mean value $\mathbf{x}_{s,0}$ and covariance matrix **C** is also denoted as:

$$\mathbf{x}_s \sim \mathcal{N}(\mathbf{x}_{s,0}, \mathbf{C}) \tag{25}$$

The covariance matrix **C** is formed by the standard deviations $\sigma_k\,,\ k = 1, \ldots, n_{xs}$ of the individual parameters $x_{s,k}\,,\ k = 1, \ldots, n_{xs}$, and by the mutual correlations $\varrho_{k,l}\,,\ k, l = 1, \ldots, n_{xs}\,,\ k \neq l$, between parameters $x_{s,k}$ and $x_{s,l}$ in the following way:

$$\mathbf{C} = \mathbf{\Sigma} \cdot \mathbf{R} \cdot \mathbf{\Sigma} \tag{26}$$

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_1 & 0 & \dots & 0 \\ 0 & \sigma_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \dots & 0 & \sigma_{n_{xs}} \end{bmatrix} \tag{27}$$

$$\mathbf{R} = \begin{bmatrix} 1 & \varrho_{1,2} & \dots & \varrho_{1,n_{xs}} \\ \varrho_{2,1} & 1 & & \vdots \\ \vdots & & \ddots & \varrho_{n_{xs}-1,n_{xs}} \\ \varrho_{n_{xs},1} & \dots & \varrho_{n_{xs},n_{xs}-1} & 1 \end{bmatrix} \tag{28}$$

$$\mathbf{C} = \begin{bmatrix} \sigma_1^2 & \sigma_1\varrho_{1,2}\sigma_2 & \dots & \sigma_1\varrho_{1,n_{sx}}\sigma_{n_{xs}} \\ \sigma_2\varrho_{2,1}\sigma_1 & \sigma_2^2 & & \vdots \\ \vdots & & \ddots & \sigma_{n_{xs}-1}\varrho_{n_{xs}-1,n_{xs}}\sigma_{n_{xs}} \\ \varrho_{n_{xs},1} & \dots & \varrho_{n_{xs},n_{xs}-1} & \sigma_{n_{xs}}^2 \end{bmatrix} \tag{29}$$

σ_k^2 is the variance of parameter $x_{s,k}$. $\sigma_k \varrho_{k,l} \sigma_l$ is the covariance of parameters $x_{s,k}$, $x_{s,l}$. For the variance, we have:

$$\sigma_k > 0 \tag{30}$$

For the correlation, we have:

$$\varrho_{k,l} = \varrho_{l,k} \tag{31}$$

$$-1 \leq \varrho_{k,l} \leq +1 \tag{32}$$

From (26)–(32) follows that the covariance matrix $\mathbf{C}$ is symmetric and positive semidefinite.

If $\varrho_{k,l} = 0$, this is denoted as the two parameters $x_{s,k}$, $x_{s,l}$ being uncorrelated. In the case of a multivariate normal distribution, "uncorrelated" is equivalent to "statistically independent." In general, "statistically independent" implies "uncorrelated."

If two parameters $x_{s,k}$, $x_{s,l}$ are perfectly correlated, then $|\varrho_{k,l}| = 1$. Perfect correlation refers to a functional relationship between the two parameters. For each value of one parameter a unique value of the other parameter corresponds, i.e. the statistical uncertainty between the two parameters disappears. In this case, the covariance matrix has a zero eigenvalue and the statistical distribution is singular, i.e. (23) cannot be written. In the following, we will assume that

$$|\varrho_{k,l}| < 1 \tag{33}$$

This means that $\mathbf{C}$ is positive definite and that the multivariate normal distribution is nonsingular. A singular distribution can be transformed into a nonsingular one [5].

For two parameters, (23) and (24) have the following form:

$$\mathbf{x}_s = \begin{bmatrix} x_{s,1} \\ x_{s,2} \end{bmatrix} \sim \mathcal{N}\left(\mathbf{x}_{s,0} = \begin{bmatrix} x_{s,0,1} \\ x_{s,0,2} \end{bmatrix}, \ \mathbf{C} = \begin{bmatrix} \sigma_1^2 & \sigma_1 \varrho \sigma_2 \\ \sigma_1 \varrho \sigma_2 & \sigma_2^2 \end{bmatrix}\right)$$

$$\begin{aligned} \mathrm{pdf}_N(x_{s,1}, x_{s,2}) \ &= \ \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\varrho^2}} \cdot \exp\left\{-\frac{1}{2(1-\varrho^2)} \cdot \left[\frac{(x_{s,1}-x_{s,0,1})^2}{\sigma_1^2}\right.\right. \\ &\left.\left. -2\varrho\frac{x_{s,1}-x_{s,0,1}}{\sigma_1}\frac{x_{s,2}-x_{s,0,2}}{\sigma_2} + \frac{(x_{s,2}-x_{s,0,2})^2}{\sigma_2^2}\right]\right\} \end{aligned} \tag{34}$$

If the two parameters are uncorrelated, i.e. $\varrho = 0$, (34) becomes:

$$\begin{aligned} \mathrm{pdf}_N(x_{s,1}, x_{s,2}) &= \frac{1}{2\pi\sigma_1\sigma_2} \cdot \exp\left\{-\frac{1}{2}\left[\frac{(x_{s,1}-x_{s,0,1})^2}{\sigma_1^2} + \frac{(x_{s,2}-x_{s,0,2})^2}{\sigma_2^2}\right]\right\} \qquad (35) \\ &= \frac{1}{\sqrt{2\pi}\sigma_1}\exp\left\{-\frac{1}{2}\frac{(x_{s,1}-x_{s,0,1})^2}{\sigma_1^2}\right\} \cdot \frac{1}{\sqrt{2\pi}\sigma_2}\exp\left\{-\frac{1}{2}\frac{(x_{s,2}-x_{s,0,2})^2}{\sigma_2^2}\right\} \\ &= \mathrm{pdf}_N(x_{s,1}) \cdot \mathrm{pdf}_N(x_{s,2}) \end{aligned}$$

If the two parameters are uncorrelated, i.e. $\varrho = 0$, and if their mean values are $x_{s,0,1} = x_{s,0,2} = 0$, and if their variance values are $\sigma_1 = \sigma_2 = 1$, (34) becomes:

$$\mathrm{pdf}_N(x_{s,1}, x_{s,2}) = \frac{1}{2\pi} \cdot \exp\left\{-\frac{1}{2}\left[x_{s,1}^2 + x_{s,2}^2\right]\right\} = \frac{1}{\sqrt{2\pi}^{n_{xs}}} \cdot \exp\left\{-\frac{1}{2}\mathbf{x}_s^T\mathbf{x}_s\right\} \tag{36}$$

The level curves of the probability density function of a normal distribution, i.e. the equidensity curves, are determined by the quadratic form (24). Therefore, the level curves of pdf_N are ellipses for $n_{xs} = 2$, ellipsoids for $n_{xs} = 3$ (Figure 21) or hyper-ellipsoids for $n_{xs} \geq 3$. In the rest of this book, three-dimensional terms like "ellipsoid" will be used regardless of the dimension.

Figure 22 illustrates different shapes of the equidensity curves of a normal probability density function in dependence of variances and correlations. Each equidensity curve is determined by the respective value of β in (24). For $\beta = 0$, the maximum value of pdf_N, given by $\frac{1}{\sqrt{2\pi}^{n_{xs}} \cdot \sqrt{\det \mathbf{C}}}$, is obtained at the mean value $\mathbf{x}_{s,0}$. Increasing values $\beta > 0$ determine concentric ellipsoids centered around the mean value $\mathbf{x}_{s,0}$ with increasing value of β and decreasing corresponding value of pdf_N.

Figure 22(a) illustrates the situation if the two parameters are positively correlated. The major axes of the ellipsoids then have a tilt in the parameter space. Note that the equidensity ellipsoids denote a tolerance region corresponding to (10). If the two parameters are uncorrelated, then the major axes of the ellipsoids correspond to the coordinate axes of the parameters, as illustrated in Figure 22(b). If uncorrelated parameters have equal variances, then the ellipsoids become spheres, as illustrated in Figure 22(c).

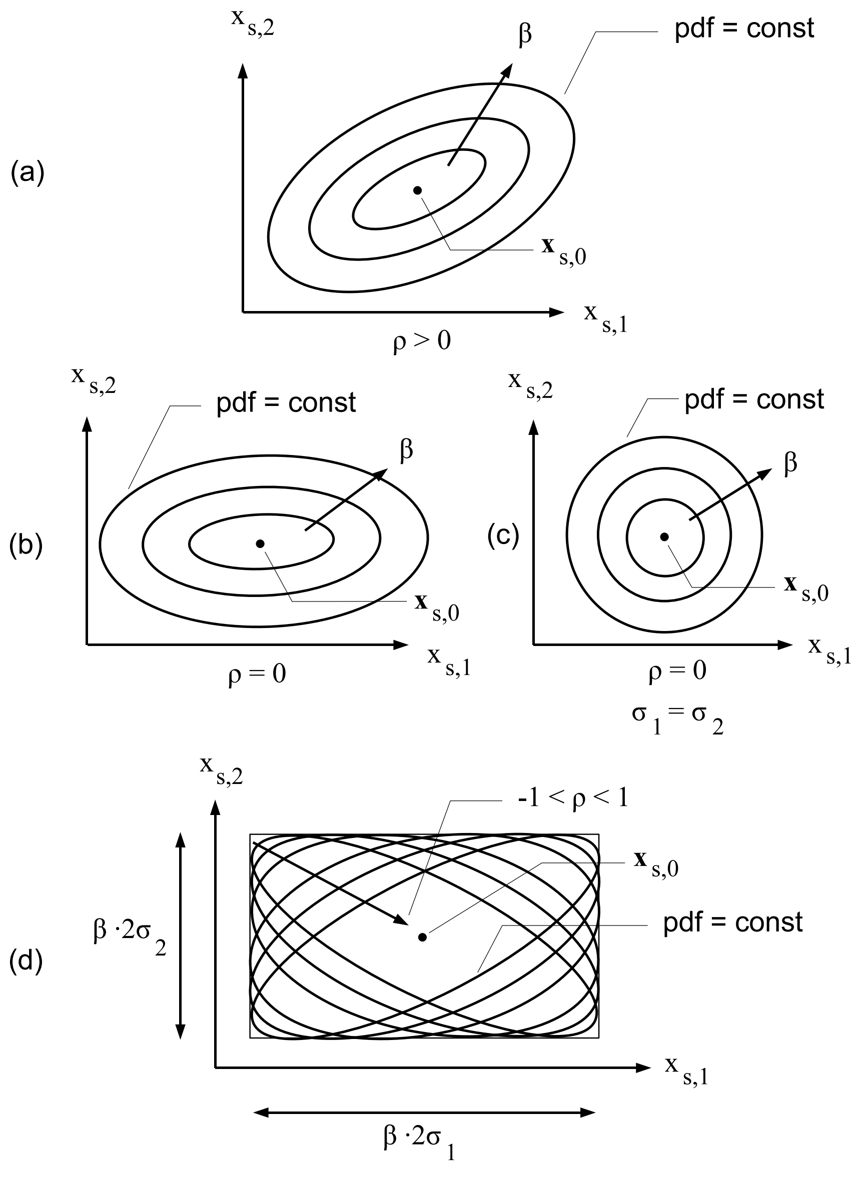


Figure 22. Different sets of level contours $\beta^2 = (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0})$ (24) of a two-dimensional normal probability density function pdf_N. *(a)* Varying β, constant positive correlation ϱ, constant unequal variances σ_1, σ_2. *(b)* Varying β, constant zero correlation, constant unequal variances. *(c)* Varying β, constant zero correlation, constant equal variances. *(d)* Varying correlation, constant β, constant unequal variances.

Figure 22(d) illustrates a set of equidensity contours for a constant value of β. A positive correlation between two parameters leads to a contraction of the equidensity ellipsoids towards the axis of parameter alterations in the same direction. A negative correlation leads to a contraction of the equidensity ellipsoids towards the axis of parameter alterations in the opposite direction. The larger the absolute value of correlation, the narrower the shape of the ellipsoid. For a correlation of ± 1, this part of the ellipsoid degenerates to the corresponding line of parameter alteration. In addition, for a constant value of β, any equidensity contour for an arbitrary value of the correlation $\varrho_{k,l}$ between two parameters lies in a bounding box around the mean value $\mathbf{x}_{s,0}$ that has a size of $\beta \cdot 2\sigma_k$ in direction of parameter $x_{s,k}$ and $\beta \cdot 2\sigma_l$ in direction of parameter $x_{s,l}$. This bounding-box-of-ellipsoids property for two statistical parameters $x_{s,k}$, $x_{s,l}$ can be formulated by:

$$\begin{aligned}\bigcup_{|\varrho_{k,l}| \leq 1} & \{\mathbf{x}_s | \; (\mathbf{x}_s - \mathbf{x}_{s,0})^T \mathbf{C}^{-1} (\mathbf{x}_s - \mathbf{x}_{s,0}) \leq \beta^2\} \\ & = \left\{\mathbf{x}_s \;\middle|\; |x_{s,k/l} - x_{s,0,\,k/l}| \leq \beta\, \sigma_{k/l}\right\} \end{aligned} \tag{37}$$

While the left set in (37) describes an ellipsoid tolerance region according to (10), the right set in (37) describes a box tolerance region according to (8). The bounding-box-of-ellipsoids property (37) says that the union of all ellipsoid tolerance regions that are obtained by varying the correlation value of two parameters for a constant β yields exactly a box tolerance region $\pm\beta\sigma_{k/l}$ around the mean value. A motivation of (37) is given in Appendix C.7.

The cumulative distribution function $\mathrm{cdf}_N(\mathbf{x}_s)$ of the multivariate normal distribution can be formulated according to (13). A more specific form analogous to (18) and (19) of the univariate normal distribution cannot be obtained. Only if the correlations are zero, then the probability density function of the multivariate normal distribution can be formulated as a product of the univariate probability density functions of the individual parameters (see (35)) and we obtain:

$$\mathbf{R} = \mathbf{I} \Rightarrow \mathrm{cdf}_N(\mathbf{x}_s) = \prod_{i=1}^{n_{xs}} \int_{-\infty}^{x_{s,k}} \mathrm{pdf}_N(t) dt = \prod_{k=1}^{n_{xs}} \mathrm{cdf}_N(x_{s,k}) \tag{38}$$

$\mathbf{I}$ denotes the identity matrix.

3.7 Transformation of Statistical Distributions

Due to the quadratic function (24) in the probability density function (23), the formulation of tolerance design tasks based on a multivariate normal distribution of parameters can be very thorough and rich in technical interpretation.

Therefore, we will assume that the statistical parameters are normally distributed.

But parameters are not normally distributed in general. For instance parameters like oxide thickness that cannot attain values below zero naturally have a skew distribution unlike the symmetric normal distribution. These parameters have to be transformed into statistical parameters whose distribution is normal. The resulting transformed parameters of course have no longer their original physical meaning. This is in fact not necessary for the problem formulation of yield optimization/design centering. The reader is asked to keep in mind that the normally distributed statistical parameters that are assumed in the rest of this book may originate from physical parameter that are distributed with another distribution.

Given two random vector variables, $\mathbf{y} \in \mathbb{R}^{n_y}$ and $\mathbf{z} \in \mathbb{R}^{n_z}$ with $n_y = n_z$, which can be mapped onto each other with a bijective function[2], i.e. $\mathbf{z}(\mathbf{y})$ and $\mathbf{y}(\mathbf{z})$, the probability density function pdf_z of random variable $\mathbf{z}$ is obtained from the probability density function pdf_y of random variable $\mathbf{y}$ by:

$$\begin{aligned} \mathrm{pdf}_z(\mathbf{z}) &= \mathrm{pdf}_y(\mathbf{y}(\mathbf{z})) \cdot \left| \det\left(\frac{\partial \mathbf{y}}{\partial \mathbf{z}^T} \right) \right| \qquad (39) \\ \mathrm{pdf}_z(z) &= \mathrm{pdf}_y(y(z)) \cdot \left| \frac{\partial y}{\partial z} \right| \end{aligned}$$

The idea behind the transformation is to keep the probability of an infinitesimal parameter interval equal while transforming it to another infinitesimal interval:

$$\mathrm{pdf}_z(\mathbf{z}) \cdot d\mathbf{z} = \mathrm{pdf}_y(\mathbf{y}) \cdot d\mathbf{y} \qquad (40)$$

This corresponds to a substitution of variable $\mathbf{y}$ in the integration over pdf_y to calculate cdf_y (13):

$$\begin{aligned} \mathrm{cdf}_y(\mathbf{y}) &= \int_{-\infty}^{y_1} \cdots \int_{-\infty}^{y_{n_y}} \mathrm{pdf}_y(\mathbf{y}) d\mathbf{y} \\ &= \int_{-\infty}^{z_1(\mathbf{y})} \cdots \int_{-\infty}^{z_{n_z}(\mathbf{y})} \mathrm{pdf}_y(\mathbf{y}(\mathbf{z})) \left| \det\left(\frac{\partial \mathbf{y}}{\partial \mathbf{z}^T} \right) \right| d\mathbf{z} \\ &= \int_{-\infty}^{z_1} \cdots \int_{-\infty}^{z_{n_z}} \mathrm{pdf}_z(\mathbf{z}) d\mathbf{z} = \mathrm{cdf}_z(\mathbf{z}) \qquad (41) \end{aligned}$$

Exercise. Prove (17) using (41).

[2] $\mathbf{z}(\mathbf{y})$ is meant to be read as "$\mathbf{z}$ evaluated at $\mathbf{y}$." This implies that $\mathbf{z}$ is a function of $\mathbf{y}$, i.e. $\mathbf{z} = \varphi(\mathbf{y})$. The function φ is not denoted separately.

Example 1. This example shows how a transformation function from one random variable to another random variable can be derived if the two probability density functions are given.

Let us consider the case that the univariate random variable z is originating from a uniform distribution, $z \sim \mathcal{U}(0,1)$ with the probability density function pdf_U:

$$\text{pdf}_U(z) = \begin{cases} 1, & 0 < z < 1 \\ 0, & \text{else} \end{cases} \tag{42}$$

We are looking for the function that maps the uniformly distributed variable z into a random variable y that is distributed according to pdf_y. Towards this, we insert (42) in (40), i.e. $1 \cdot dz = \text{pdf}_y(y) \cdot dy$, and obtain:

$$y = \text{cdf}_y^{-1}(z) \tag{43}$$

(43) says that a uniformly distributed random variable z is transformed into a random variable y that is distributed according to pdf_y through the inverse cumulative distribution function of y. If pdf_y is given, this is done by

- solving $\int_{-\infty}^{y} \text{pdf}_y(t)dt$ to compute $\text{cdf}_y(y)$ and then
- computing the inverse cdf_y^{-1}.

Example 2. This example shows how the probability density function pdf_z of random variable z can be derived if the probability density function pdf_y of random variable y and the transformation function $y \mapsto z$ are given.

Given as an example are a random variable y originating from a normal distribution (44) and an exponential function (45) that maps y onto z:

$$\text{pdf}_y(y) = \frac{1}{\sqrt{2\pi} \cdot \sigma} \, \text{e}^{-\frac{1}{2}\left(\frac{y-y_0}{\sigma}\right)^2} \tag{44}$$

$$z = \text{e}^y \tag{45}$$

We are looking for the probability density function $\text{pdf}_z(z)$. As the function $z = \text{e}^y$ is bijective w.r.t. $\mathbb{R} \mapsto \mathbb{R}^+$, (39) is applied:

$$\text{pdf}_z(z) = \text{pdf}_y(y(z)) \cdot \left|\frac{\partial y}{\partial z}\right| \quad \text{with} \quad y = \ln z \quad \text{and} \quad \left|\frac{\partial y}{\partial z}\right| = \frac{1}{z} \quad \text{yields}$$

$$\text{pdf}_z(z) = \frac{1}{\sqrt{2\pi} \cdot \sigma \cdot z} \, \text{e}^{-\frac{1}{2}\left(\frac{\ln z - y_0}{\sigma}\right)^2} \tag{46}$$

(46) denotes the pdf of a lognormal distribution. As can be seen from the calculations above, the logarithm $y = \ln x$ of a lognormally distributed random

variable z is normally distributed. As $z > 0$, the lognormal distribution may be a candidate for modeling the distribution of a random variable that does not attain values below zero. Based on the transformation (45), a normally distributed random variable can be obtained.

Example 3. This example is of the same type as Example 2. Given are a random variable y originating from a standard normal distribution (47) and a quadratic function (48) that maps y onto z:

$$\mathrm{pdf}_y(y) = \frac{1}{\sqrt{2\pi}}\,\mathrm{e}^{-\frac{1}{2}y^2} \tag{47}$$

$$z = y^2 \tag{48}$$

We are looking for the probability density function $\mathrm{pdf}_z(z)$. The function $z = y^2$ is not bijective, but surjective w.r.t. $\mathbb{R} \mapsto \mathbb{R}^+$. In this case we can add the probabilities of all infinitesimal intervals dy that contribute to the probability of an infinitesimal interval dz. Two values, $y^{(1)} = +\sqrt{z}$ and $y^{(2)} = -\sqrt{z}$, lead to the same value z and hence two infinitesimal intervals in y contribute to the probability of dz:

$$\mathrm{pdf}_z(z) \cdot dz = \mathrm{pdf}_y\left(y^{(1)}\right) \cdot dy^{(1)} + \mathrm{pdf}_y\left(y^{(2)}\right) \cdot dy^{(2)} \tag{49}$$

This leads us to:

$$\begin{aligned}\mathrm{pdf}_z(z) &= \mathrm{pdf}_y\left(y^{(1)}(z)\right)\left|\frac{\partial y^{(1)}}{\partial z}\right| + \mathrm{pdf}_y\left(y^{(2)}(z)\right)\left|\frac{\partial y^{(2)}}{\partial z}\right| \\ &= \frac{1}{\sqrt{2\pi}}\,\mathrm{e}^{-\frac{1}{2}z} \cdot \left(\left|\frac{1}{2}z^{-\frac{1}{2}}\right| + \left|-\frac{1}{2}z^{-\frac{1}{2}}\right|\right) \\ &= \frac{1}{\sqrt{2\pi}} \cdot z^{-\frac{1}{2}} \cdot \mathrm{e}^{-\frac{1}{2}z} \end{aligned} \tag{50}$$

(50) denotes the probability density function of a χ^2(chi-square)-distribution with one degree of freedom. The χ^2 distribution can be applied to describe the probability of ellipsoidal tolerance regions determined by β (24).

The transformation of statistical distributions and of the corresponding random variables is also important for statistical estimations, for instance by a Monte-Carlo analysis. Towards that, a statistical sample, i.e. a finite set of sample elements, is generated, which is originating from a certain statistical distribution. In the next section, the generation of sample elements from a normal distribution will be described.

3.8 Generation of Normally Distributed Sample Elements

The generation of normally distributed sample elements[3] $\mathbf{x}_s \sim \mathcal{N}(\mathbf{x}_{s,0}, \mathbf{C})$ is done in three steps:

- Create a vector $\mathbf{z}$ with vector elements that are independently uniformly distributed, i.e. $z_k \sim U(0,1)$, $k = 1, \ldots, n_{xs}$. This step is based on methods to create pseudo-random sequences of real numbers between 0 and 1. This is implemented for instance by a linear feedback shift register, where the XOR operation of several bits is fed back to the input. Another approach is a multiple recursive generator $z_k = w_n/\nu$, $w_n := (a_1 w_{n-1} + \ldots + a_\mu w_{n-\mu}) \bmod \nu$, of order μ and a prime modulus ν. Programming libraries usually contain functions for the generation of uniformly distributed pseudo-random numbers.

- Map $\mathbf{z}$ onto a random variable $\mathbf{y}$ that is distributed under a standardized normal distribution, i.e. $\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $\mathbf{y} \in \mathbb{R}^{n_{xs}}$. This is done as in Example 1 of the previous section. According to (43), the value z_k obtained from a uniform distribution is mapped onto a value y_k from a univariate standard normal distribution by $y_k = cdf_N^{-1}(z_k)$. cdf_N^{-1} is evaluated through functions like φ_0 (18) or erf (19), which are contained in programming libraries. The uniformly distributed values z_k, $k = 1, \ldots, n_{xs}$, are statistically pseudo-independent, and the values y_k, $k = 1, \ldots, n_{xs}$, are collected in a random vector $\mathbf{y}$ of normally distributed, pseudo-uncorrelated vector elements.

- Map $\mathbf{y}$ into a random variable $\mathbf{x}_s$ that is distributed under a normal distribution, i.e. $\mathbf{x}_s \sim \mathcal{N}(\mathbf{x}_{s,0}, \mathbf{C})$. In this third step, $\mathbf{y}$ is transformed into a random variable $\mathbf{x}_s \sim \mathcal{N}(\mathbf{x}_{s,0}, \mathbf{C})$ by a linear transformation:

$$\mathbf{x}_s = \mathbf{A} \cdot \mathbf{y} + \mathbf{b} \tag{51}$$

The unknown coefficients $\mathbf{A}$ and $\mathbf{b}$ can be derived by the requirements that the expectation value of $\mathbf{x}_s$ is $\mathbf{x}_{s,0}$, and that the variance of $\mathbf{x}_s$ is $\mathbf{C}$:

$$\mathrm{E}\{\mathbf{x}_s\} = \mathbf{x}_{s,0} \tag{52}$$

$$\mathrm{V}\{\mathbf{x}_s\} = \mathbf{C} \tag{53}$$

Inserting (51) in (52) and (53) and applying (A.12) and (A.13) we obtain:

$$\mathrm{E}\{\mathbf{x}_s\} = \mathbf{A} \cdot \mathrm{E}\{\mathbf{y}\} + \mathbf{b} = \mathbf{b} \tag{54}$$

$$\mathrm{V}\{\mathbf{x}_s\} = \mathbf{A} \cdot \mathrm{V}\{\mathbf{y}\} \cdot \mathbf{A}^T = \mathbf{A} \cdot \mathbf{A}^T \tag{55}$$

[3] A sample element is represented by a vector of parameter values $\mathbf{x}$. It refers to the stochastic event of an infinitesimal range of values $d\mathbf{x}$ around $\mathbf{x}$.

Comparison of (52) and (53) with (54) and (55) finally yields

$$\mathbf{y} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad \mapsto \quad \mathbf{x}_s \sim \mathcal{N}(\mathbf{x}_{s,0}, \mathbf{C}), \quad \mathbf{C} = \mathbf{A} \cdot \mathbf{A}^T \tag{56}$$

$$\mathbf{x}_s = \mathbf{A} \cdot \mathbf{y} + \mathbf{x}_{s,0} \tag{57}$$

$$\mathbf{y} = \mathbf{A}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \tag{58}$$

Transformation formulas (57) and (58) can be verified by showing that they transform the probability density function of the multivariate standard normal distribution (36) into the probability density function of the general multivariate normal distribution (23)

Exercise. Apply (39) to verify transformation formulas (57) and (58).

In the transformation (57) and (58), no specific assumption about the form of the matrix $\mathbf{A}$ is included. Therefore, not only a Cholesky decomposition, which yields a triangular matrix, can be applied, but also an eigenvalue decomposition. As $\mathbf{C}$ is positive definite, we obtain:

$$\mathbf{C} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T = \mathbf{V}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{V}^T, \tag{59}$$

where $\mathbf{V}$ is the matrix of eigenvectors and where $\mathbf{\Lambda}^{\frac{1}{2}}$ is the diagonal matrix of the roots of the eigenvalues. A comparison with (56) leads to:

$$\mathbf{A} = \mathbf{V}\mathbf{\Lambda}^{\frac{1}{2}} \tag{60}$$

An eigenvalue decomposition is to be preferred if the number of statistical parameters is not too large because of the numerical ill-conditioning of $\mathbf{C}$ due to highly correlated parameters.

3.9 Global and Local Parameter Tolerances

The modeling of the statistical variations of an integrated circuit manufacturing process is a complex task. On circuit and architecture level, two types of parameter tolerances are assumed, global and local tolerances.

Global parameter tolerances are due to chip-to-chip (“inter-chip”) and wafer-to-wafer fluctuations of the manufacturing process. They are modeled as parameter variations that affect all transistors of a circuit in the same way, hence the notation global. Typical examples of statistical parameters with global tolerances are oxide thickness or channel length reduction.

Besides global parameter tolerances, there are local parameter tolerances that are due to variations within a chip (“intra-chip”) and that affect transistors individually.

Local variations on circuit level are usually assumed as statistically independent variations of individual transistors. Local variations decrease with increasing gate area and increasing distance between transistors [95]. They lead to different behaviors of transistors in a circuit, a so-called mismatch. But analog circuits are formed of transistor pairs like current mirrors requiring an identical behavior of the transistors in a pair and are therefore very sensitive to mismatch. Local variations have always been an issue in analog design. But as the device dimensions become smaller and smaller, the significance of local variations compared to global variations grows critical for digital circuits as well. The statistical variation of critical paths' delays for instance are strongly depending on the local and global process variations and their effect on the correlation of gate delays.

A typical example of statistical parameters with local tolerances are the transistor threshold voltages. Local variations lead to an increase in the number of statistical parameters that corresponds to the number of transistors in a circuit. In a circuit with 100 transistors for instance there will be one global threshold voltage $V_{th,glob}$ that affects all transistors equally and 100 local threshold voltages $V_{th,loc,i}$, $i = 1, \ldots, 100$ that affect each transistor individually and independently from the others.

According to the assumptions that global statistical parameters are not correlated with local statistical parameters and that local statistical parameters are not correlated with each other, the covariance matrix has the following structure:

$$\mathbf{x}_s = \begin{bmatrix} \mathbf{x}_{glob} \\ \mathbf{x}_{loc} \end{bmatrix}, \; \mathbf{C} = \begin{bmatrix} \mathbf{C}_{glob} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_{loc} \end{bmatrix}, \; \mathbf{\Sigma}_{loc} = \mathrm{diag}(\sigma_{loc,1} \ldots \sigma_{loc,n_{loc}}) \quad (61)$$

Note that the actual transistor parameter at the interface to the circuit simulator may be a sum of a deterministic design parameter component $x_{d,k}$, of a global statistical parameter component $x_{s,glob,l}$, and of a local statistical parameter component $x_{s,loc,m}$.

3.10 Performance Features

Corresponding to the definition of parameters, performance features shall be used specifically for those quantities that are output of the numerical simulation.

Definition: Performance Features *are variable quantities of an analog system or an analog component that are output quantities of numerical simulation. The elements f_i of the performance vector $\mathbf{f} = [f_1 \ldots f_i \ldots f_{n_f}]^T \in \mathbb{R}^{n_f}$ denote the values of the respective performance features, whose names are addressed through the index i.*

Examples of performance features of an operational amplifier are gain, phase margin, slew rate, power supply rejection ratio.

Performance features are determined by the parameters. For a certain parameter vector, a certain vector of performance features is obtained:

$$\mathbf{x} \mapsto \mathbf{f} \tag{62}$$

As mentioned in Note 2 in Section 3.7, we do not explicitly denote the function that maps a parameter vector onto a performance vector: $\varphi : \mathbf{x} \rightarrow \mathbf{f}$. Instead, $\mathbf{f}(\mathbf{x})$ is meant to denote "$\mathbf{f}$ evaluated at $\mathbf{x}$," which implies that such a function exists.

The performance features $\mathbf{f}$ for given parameters $\mathbf{x}$ are usually evaluated by means of numerical simulation.

3.11 Numerical Simulation

It is important to note that in general the circuit behavior is not formulated explicitly as in the RC circuit in Chapter 2, but is described by nonlinear differential algebraic equations (DAEs),

$$\varphi\left(t, \mathbf{x}, \mathbf{u}_n(\mathbf{x}, t), \mathbf{i}_Z(\mathbf{x}, t), \dot{\mathbf{q}}(\mathbf{u}_n, \mathbf{i}_Z, \mathbf{x}, t), \dot{\boldsymbol{\phi}}(\mathbf{u}_n, \mathbf{i}_Z, \mathbf{x}, t), \right) = \mathbf{0}, \tag{63}$$

where t denotes the time, $\mathbf{x}$ the parameters, $\mathbf{u}_n$ the node voltages, $\mathbf{i}_Z$ the currents in Z-branches, $\mathbf{q}$ the capacitor charges, $\boldsymbol{\phi}$ the inductor fluxes, and $\dot{}$ the derivative with respect to time. Performance evaluation in general is a map of a certain parameter vector $\mathbf{x}$ onto the resulting performance feature vector $\mathbf{f}$, which is carried out in two stages. First, numerical simulation [97, 75, 121, 101, 89] by means of numerical integration methods maps a parameter vector onto the node voltages and Z-branch currents:

$$\mathbf{x} \mapsto \mathbf{u}_n, \mathbf{i}_Z \tag{64}$$

In case of a DC simulation, the resulting node voltages and Z-branch currents are constant values. In case of an AC simulation, they are a function of frequency, and in case of a transient (TR) simulation, they are a function of time. These functions are given in the original sense of a relation as a finite set of $(\mathbf{x}, [\mathbf{u}_n^T \ \mathbf{i}_Z^T]^T)$-elements. In a second post-processing step, the node voltages and Z-branch currents are mapped onto the actual performance features by problem-specific operations:

$$\mathbf{u}_n, \mathbf{i}_Z \mapsto \mathbf{f} \tag{65}$$

Depending on the type of simulation, we obtain corresponding DC performance features like for instance DC power consumption or transistor operating voltages, AC performance feature like gain or phase margin, and TR performance features like slew rate.

Numerical simulation is computationally extremely expensive, a single simulation can even take days. Contrary to our simple illustrative example in Chapter 2, general analog performance functions

- can only be established element-wise $(\mathbf{x}, \mathbf{f})$ for selected parameter vectors,
- and the calculation of each function element costs minutes to hours CPU time.

Algorithms for analog optimization must therefore very carefully spend numerical simulations, just as oil exploration has to carefully spend the expensive test drillings.

In order to avoid the high computational cost of numerical simulation, performance models or macromodels for a cheaper function evaluation would be welcomed. Performance models can be obtained through numerical methods called response surface modeling [31, 39, 3, 76, 37, 124, 16, 125, 127, 43, 64, 78, 17, 4, 126, 14, 29] or through analytical methods like symbolic analysis. Macromodels can be obtained by simplified DAEs [22], by symbolic analysis [108, 112, 113, 30, 107, 106, 32, 117, 67, 120, 119, 44, 18, 114, 15, 51, 50, 53, 52], by equivalent circuits [49, 20], or by behavioral models [128, 54, 65, 71, 72, 82]. However, the generation of such models requires a large number of numerical simulations. Only if the resulting models are applied often enough the setup cost are worthwhile. We are not aware of investigations of the break-even where pre-optimization modeling undercuts the cost of optimization on direct numerical simulation.

3.12 Performance-Specification Features

The performance features are subject to constraints that have to be satisfied. These constraints are called performance-specification features and are derived from the system performance that has been contracted with the customer. The verification or validation of a circuit with regard to the performance specification may be required at different stages of the design and test process, for instance after structural synthesis prior to the layout design phase, or after production test prior to the delivery.

Definition. *A* **performance-specification feature** *is an inequality of the form:*

$$\varphi_{PSF,\mu}(\mathbf{f}) \geq 0 \tag{66}$$

The **performance specification** *consists of performance-specification features, which determine the acceptable performance region:*

$$\varphi_{PSF,\mu}(\mathbf{f}) \geq 0, \ \ \mu = 1, \ldots, n_{PSF} \tag{67}$$

If the values of the performance features satisfy all performance-specification features, i.e. are inside the acceptable performance region, then a circuit is classified as "in full working order."

The prevalent form of performance-specification features are upper boundary values f_U or lower boundary values f_L that the performance may not exceed or fall below. In this case, a performance-specification feature has the following form,

$$f_i - f_{L,i} \geq 0 \text{ or } f_{U,i} - f_i \geq 0 \tag{68}$$

and the performance specification is:

$$f_i - f_{L,i} \geq 0 \wedge f_{U,i} - f_i \geq 0, \; i = 1, \ldots, n_f \tag{69}$$

The set of all boundary values determines a performance acceptance region A_f:

$$\begin{aligned} A_f &= \{\mathbf{f} \mid \mathbf{f}_L \leq \mathbf{f} \leq \mathbf{f}_U\} \\ &= \{\mathbf{f} \mid f_i \geq f_{L,i} \wedge f_i \leq f_{U,i} \;, \;\; i = 1, \ldots, n_f\} \end{aligned} \tag{70}$$

The vector inequality is defined as in (12). For each performance feature f_i, either a lower bound $f_{U,i}$ or an upper bound $f_{L,i}$ or both a lower and an upper bound may be specified. If a bound is not specified, the value of the corresponding bound in (70) would be $f_{L,i} \rightarrow -\infty$ or $f_{u,i} \rightarrow \infty$.

(70) has the same form as (8) and the acceptable performance region is a box region as illustrated in Figure 19(a). Performance-specification features in general may acquire the forms given in (8)–(11) and the corresponding box, polytope, ellipsoid or general nonlinear geometries, or combinations of these forms.

Chapter 4

ANALOG SIZING TASKS

4.1 Sensitivity-Based Analysis

Continuous optimization of analog circuits assumes a certain "smoothness" of the optimization objectives in the parameters, i.e. continuously differentiable objective functions with a certain degree of continuous differentiability [56]. For such functions, the concept of sensitivity can be applied. Basically, the sensitivity is described by the first-order derivatives of objectives, for instance the performance features $\mathbf{f}$, with respect to parameters $\mathbf{x}$:

$$\begin{aligned}
\mathbf{S} &= \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}^T}\right|_{\mathbf{x}} = \nabla \mathbf{f}(\mathbf{x}^T) = \mathbf{J}, \qquad \mathbf{S} \in \mathbb{R}^{n_f \times n_x} \\
\mathbf{S} &= \left.\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_{n_x}} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_{n_x}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_{n_f}}{\partial x_1} & \frac{\partial f_{n_f}}{\partial x_2} & \cdots & \frac{\partial f_{n_f}}{\partial x_{n_x}} \end{bmatrix}\right|_{\mathbf{x}} \\
&= \begin{bmatrix} \nabla f_1(x_1) & \nabla f_1(x_2) & \dots & \nabla f_1(x_{n_x}) \\ \nabla f_2(x_1) & \nabla f_2(x_2) & \dots & \nabla f_2(x_{n_x}) \\ \vdots & \vdots & \ddots & \vdots \\ \nabla f_{n_f}(x_1) & \nabla f_{n_f}(x_2) & \dots & \nabla f_{n_f}(x_{n_x}) \end{bmatrix} \\
&= \begin{bmatrix} \nabla f_1(\mathbf{x})^T \\ \nabla f_2(\mathbf{x})^T \\ \vdots \\ \nabla f_{n_f}(\mathbf{x})^T \end{bmatrix} = \begin{bmatrix} \nabla \mathbf{f}(x_1) & \nabla \mathbf{f}(x_2) & \dots & \nabla \mathbf{f}(x_{n_x}) \end{bmatrix} \qquad (71)
\end{aligned}$$

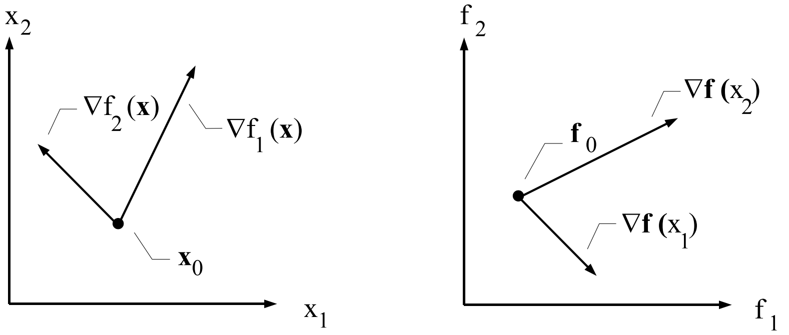


Figure 23. Sensitivity matrix in the parameter and performance space.

The matrix in (71) is called sensitivity matrix $\mathbf{S}$, gradient $\nabla\mathbf{f}(\mathbf{x}^T)$, or Jacobian matrix $\mathbf{J}$. The i, kth element of $\mathbf{S}$ denotes the partial derivative of the performance feature f_i with respect to the parameter x_k: $\left.\frac{\partial f_i}{\partial x_k}\right|_{\mathbf{x}} = \nabla f_i(x_k)$.[1] The ith row $\nabla f_i(\mathbf{x})^T$ denotes the gradient of performance feature f_i with respect to the parameters $\mathbf{x}$, the kth column $\nabla\mathbf{f}(x_k)$ denotes the gradient of the performance $\mathbf{f}$ with respect to parameter x_k.

The sensitivity matrix represents the first-order change of the performance feature values $\Delta\mathbf{f}$ in dependence of a change in the parameters $\Delta\mathbf{x}$:

$$\Delta\mathbf{f} = \mathbf{S}\cdot\Delta\mathbf{x} = \sum_{k=1}^{n_x} \nabla\mathbf{f}(x_k)\cdot\Delta x_k \tag{72}$$

The sensitivity matrix allows to separate the effects of a single-parameter alteration or a multiple-parameter alteration on a single or multiple performance features and is therefore of eminent significance for the analysis and optimization of analog circuits. Figure 23 illustrates the sensitivity matrix for an example with two parameters and two performance features.

On the left side, the directions of steepest ascent for each of the two performance features are shown. On the right side, the effects of alterations in either of the two parameters on the performance feature values are shown.

In the following, some aspects of a sensitivity-based analysis are formulated.

[1] Please note that the Nabla symbol does not denote the Nabla operator here. In the abbreviated notation of the partial derivative with the Nabla sign the parameters in brackets denote the parameters with regard to which the partial derivative is calculated. Other parameters, which are held constant at certain values, are not mentioned explicitly in this notation, but they determine the result value as well.

4.1.1 Similarity of Performance Features

If the two gradients $\nabla f_i(\mathbf{x})$ and $\nabla f_j(\mathbf{x})$ are orthogonal to each other, i.e. linearly independent, an altered parameter vector in the direction of one of the gradients will only affect one performance feature, while the other one remains unchanged. Linear independence of the two gradients $\nabla f_i(\mathbf{x})$ and $\nabla f_j(\mathbf{x})$ hence means that the two corresponding performance features can be tuned independently from each other. We could call them minimum similar.

If on the other hand, the two gradients $\nabla f_i(\mathbf{x})$ and $\nabla f_j(\mathbf{x})$ are parallel to each other, i.e. linearly dependent, changes in the two corresponding performance features are inseparable from each other to the first order. We could call the two performance features maximum similar.

Similarity of two performance features f_i, f_j is a measure of the similarity of their reaction on alterations in the parameter vector. It can be described through the angle $\phi(f_i, f_j)$ between their gradients with respect to the parameters:

$$\cos\phi(f_i, f_j) = \frac{\nabla f_i(\mathbf{x})^T \cdot \nabla f_j(\mathbf{x})}{\|\nabla f_i(\mathbf{x})\| \cdot \|\nabla f_j(\mathbf{x})\|}, \quad -1 \leq \cos\phi(f_i, f_j) \leq +1 \qquad (73)$$

The smaller the angle between the gradients of the two performance features with respect to all parameters is, the more similar is the effect of parameter alteration on them. A similarity of 0 corresponds to linear independence and independent adjustment through parameter alteration. A similarity of ±1 corresponds to linear dependence and inseparable reaction on parameter alterations.

The concept of similarity corresponds to the statistical concept of correlation.

4.1.2 Similarity of Parameters

The similarity of two parameters x_k, x_l is described by the cosine of the angle between the gradients of the performance vector with respect to these two parameters:

$$\cos\phi(x_k, x_l) = \frac{\nabla \mathbf{f}(x_k)^T \cdot \nabla \mathbf{f}(x_l)}{\|\nabla \mathbf{f}(x_k)\| \cdot \|\nabla \mathbf{f}(x_l)\|}, \quad -1 \leq \cos\phi(f_i, f_j) \leq +1 \qquad (74)$$

The smaller the angle between these gradients is, the more similar are the two corresponding parameters and the more similar is their effect on all performance features.

4.1.3 Significance of Parameters

The significance of a parameter x_k corresponds to the amount of change in the values of all performance features, $\Delta\mathbf{f}(\Delta x_k)$, that it effects:

$$\Delta\mathbf{f}(\Delta x_k) = \mathbf{S} \cdot \Delta x_k \cdot \mathbf{e}_k = \nabla \mathbf{f}(x_k) \cdot \Delta x_k \qquad (75)$$

$\mathbf{e}_k$ is a vector with a 1 in the kth position and zeros in the other positions.

If we assume the same alteration Δx for all parameters, then a ranking of the parameters according to their significance is obtained by comparing the lengths of the performance gradients with respect to the individual parameters:

$$x_k \text{ more significant than } x_l \Leftrightarrow \|\nabla \mathbf{f}(x_k)\| > \|\nabla \mathbf{f}(x_l)\| \tag{76}$$

4.1.4 Adjustability of Performance Features

Analogously, the adjustability of a performance feature f_i is based on the amount of change in its value that it shows for an alteration of parameter values:

$$\Delta f_i(\Delta \mathbf{x}) = \nabla f_i(\mathbf{x})^T \cdot \Delta \mathbf{x} \tag{77}$$

By inspection of (77), we can see that the maximum amount of change in the value of f_i is obtained, if the parameters are altered according to the gradient of the performance feature:

$$\max_{\Delta \mathbf{x}} \Delta f_i(\Delta \mathbf{x}) \quad \rightarrow \quad \Delta \mathbf{x}_{max} = \nabla f_i(\mathbf{x}) \ , \ \ \Delta f_{i,max} = \|\nabla f_i(\mathbf{x})\|_2^2 \tag{78}$$

A ranking of the performance features according to their adjustability is obtained by comparing the lengths of the gradients of the individual performance features:

$$f_i \text{ more adjustable than } f_j \Leftrightarrow \|\nabla f_i(\mathbf{x})\| > \|\nabla f_j(\mathbf{x})\| \tag{79}$$

4.1.5 Multiple-Objective Behavior

The maximum increase $\Delta f_{i,max}$ in the value of performance feature f_i is obtained if the parameters are changed according to the corresponding gradient of this performance. The resulting change in all other performance features results from inserting the corresponding gradient of this performance, $\nabla f_i(\mathbf{x})$, in (72):

$$\Delta \mathbf{f}|_{\Delta f_{i,max}} = \mathbf{S} \cdot \nabla f_i(\mathbf{x}) \tag{80}$$

In this way, we can analyze how performance features change their values if one them is optimized. This is a first step towards optimization of multiple objectives.

4.1.6 Exercise

Given is the sensitivity matrix for two performance features f_1, f_2 and two parameters x_1, x_2 at a certain point $\mathbf{x}_0$ (Figure 23):

$$\mathbf{S} = \begin{bmatrix} 1 & 2 \\ -1 & 1 \end{bmatrix} \tag{81}$$

Analyze the design situation regarding similarity, significance, adjustability and multiple-objective behavior.

4.1.7 Sensitivity-Based Analysis of Tolerance Objectives

Please note that the given formulas for analyzing multiple-performance/ multiple-parameter design tasks can be applied for any type of objective. In nominal design, the nominal performance feature values are the objectives. Tolerance design's objectives are worst-case performance feature values and worst-case distances.

4.2 Performance-Sensitivity Computation

A sensitivity-based analysis requires the availability of the gradients of the respective objectives.

The performance-to-parameter sensitivities $\nabla\mathbf{f}(\mathbf{x}^T)$ are computed in two ways, depending on the capabilities of the numerical simulator.

4.2.1 Simulator-Internal Computation

Some simulators provide the sensitivities $\nabla\mathbf{u}_n(\mathbf{x}^T)$, $\nabla\mathbf{i}_Z(\mathbf{x}^T)$, from which $\nabla\mathbf{f}(\mathbf{x}^T)$ can be computed by applying the chain rule of differentiation:

$$\nabla\mathbf{f}(\mathbf{x}^T) = \nabla\mathbf{f}(\mathbf{u}_n^T) \cdot \nabla\mathbf{u}_n(\mathbf{x}^T) + \nabla\mathbf{f}(\mathbf{i}_Z^T) \cdot \nabla\mathbf{i}_Z(\mathbf{x}^T) \tag{82}$$

The formulas to compute $\nabla\mathbf{u}_n(\mathbf{x}^T)$ and $\nabla\mathbf{i}_Z(\mathbf{x}^T)$ require the corresponding parts in the transistor models. This increases the transistor modeling effort, as not only the transistor equations but also their first derivatives have to be provided.

The formulas to compute $\mathbf{f}(\mathbf{u}_n^T)$ and $\nabla\mathbf{f}(\mathbf{i}_Z^T)$ are part of the postprocessing operations that have to include the corresponding first derivatives.

The main effort is the a-priori calculation of the formulas for the first derivatives. This is done once for each transistor class of the underlying production technology and once for the considered circuit class.

The simulation overhead for sensitivity computation remains relatively small. It amounts to about 10% of the CPU time of one simulation for the computation of the sensitivity of all performance features with respect to one parameter.

4.2.2 Finite-Difference Approximation

Many simulators focus on providing an interface to apply user-defined transistor and device models. User-defined device models mostly do not include the derivatives with respect to parameters. In this frequent case, a simulator-internal sensitivity computation is not possible.

Instead, a sensitivity computation can be done by a finite-difference approximation of the gradient, which can be interpreted geometrically as a secant to the performance function at the current parameter vector. Two simulations are performed, one at the current parameter vector $\mathbf{x}$, one at a parameter vector where component x_k is altered by a certain amount Δx_k. The gradient with respect to parameter x_k is approximately the quotient of the performance difference and the parameter difference:

$$\nabla \mathbf{f}(x_k) \approx \frac{\mathbf{f}(\mathbf{x} + \Delta x_k \cdot \mathbf{e}_k) - \mathbf{f}(\mathbf{x})}{\Delta x_k} \tag{83}$$

The finite-difference approximation has no a-priori overhead, as no analytical first-order derivatives are calculated. But the simulation overhead is one simulation, i.e. 100% of the CPU time of one simulation, for computing the sensitivity of all performance features with respect to one parameter.

The choice of Δx_k is critical. It has to be large enough to surmount numerical noise and small enough to compare to the gradient.

4.3 Scaling Parameters and Performance Features

In a multiple-parameter/multiple-objective optimization problem like analog circuit design, an appropriate scaling of the parameters and performance features is crucial for the effectiveness of the solution algorithms. Missing or improper scaling leads to a deterioration in the numerical problem condition and probable failure of the solution methods. Scaling means a transformation of parameters, performance features or other design variables:

$$\mathbf{x} \mapsto \mathbf{x}', \quad \mathbf{f} \mapsto \mathbf{f}' \tag{84}$$

4.3.1 Scaling to a Reference Point

Huge differences in orders of magnitudes of physical variables like capacitors in 10^{-12}F and frequencies in 10^{6}Hz have to be eliminated. This suggests to scale a variable with respect to a reference point:

$$x'_k = \frac{x_k}{x_{RP,i}}, \quad k = 1, \ldots, n_x, \quad f'_i = \frac{f_i}{f_{RP,i}}\ , \quad i = 1, \ldots, n_f \tag{85}$$

Examples for reference points are lower bounds, upper bounds or initial values:

$$x_{RP} \in \{x_L, x_U, x_0, \ldots\}\ , \quad f_{RP} \in \{f_L, f_U, f_0, \ldots\} \tag{86}$$

4.3.2 Scaling to the Covered Range of Values

Another approach is to scale the design variables according to the range of values they cover during the design [56]:

$$x'_k = \frac{x_k - x_{RP,L,k}}{x_{RP,U,k} - x_{RP,L,k}}\ , \quad i = 1, \ldots, n_x$$

$$f'_i = \frac{f_i - f_{RP,L,i}}{f_{RP,U,i} - f_{RP,L,i}}\ , \quad i = 1, \ldots, n_f \tag{87}$$

The scaled variables according to (87) will be in the interval between 0 and 1: $x'_k, f'_i \in [0,1]$. The values $x_{RP,L,k}, x_{RP,U,k}, f_{RP,L,i}, f_{RP,U,i}$ have to be chosen very carefully. The performance specification (70) or tolerances range of parameters (8) may be helpful at this point. If only an upper or a lower bound of a variable is available, the proper choice of the opposite bound is very difficult. Initial guesses are used in these cases, which have to be tightened or relaxed according to the progress of the optimization process.

4.3.3 Scaling by Affine Transformation

A more general approach is to not only scale variables separately, but to apply affine transformations. An example for such a transformation is to scale the statistical parameters according to (56)–(58):

$$\mathbf{x}'_s = \mathbf{A}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0})\,, \quad \mathbf{C} = \mathbf{A} \cdot \mathbf{A}^T \tag{88}$$

Statistical parameters scaled according to (88) are independent from each other and have most probably values between -3 and 3. The resulting covariance matrix is the unity matrix and has the best condition number, i.e. 1.

4.3.4 Scaling by Equalization of Sensitivities

An extended approach to the scaling of variables is the inclusion of sensitivity. The linear model (72) can be interpreted as a first-order approach to map a designated change in the performance feature values $\Delta\mathbf{f}$ onto the required parameter changes $\Delta\mathbf{x}$:

$$\Delta\mathbf{f}\ , \quad \mathbf{S} \cdot \Delta\mathbf{x} = \Delta\mathbf{f} \quad \rightarrow \quad \Delta\mathbf{x} \tag{89}$$

The solution of (89) requires the solution of a system of linear equations that may be rectangular and rank-deficient. The computational accuracy of the solution is determined by the condition number of the system matrix, $\mathrm{cond}(\mathbf{S}) = \|\mathbf{S}^{-1}\| \cdot \|\mathbf{S}\|$. Gradients that differ very much cause an ill-conditioned sensitivity matrix. An improvement is to equalize the sensitivities, which in turn leads to a corresponding scaling of performance features and parameters.

There are two approaches to equalize the sensitivity matrix $\mathbf{S}$.

The first approach is to equalize the rows of $\mathbf{S}$ to an equal length of 1:

$$\begin{aligned} \underbrace{\frac{\Delta f_i}{\|\nabla f_i(\mathbf{x})\|}} &= \underbrace{\frac{\nabla f_i(\mathbf{x})^T}{\|\nabla f_i(\mathbf{x})\|}} \cdot \Delta\mathbf{x} \\ \Delta f'_i &= \nabla f'_i(\mathbf{x})^T \cdot \Delta\mathbf{x}, \quad \|\nabla f'_i(\mathbf{x})\| = 1 \end{aligned} \tag{90}$$

(90) shows that normalization of row i of the sensitivity matrix to length 1 corresponds to scaling the performance feature f_i by the norm of row i.

The second approach is to equalize the columns of $\mathbf{S}$ to an equal length of 1:

$$\begin{aligned}\Delta \mathrm{f} &= \sum_k \underbrace{\frac{\nabla \mathbf{f}(x_k)}{\|\nabla \mathbf{f}(x_k)\|}}_{} \cdot \underbrace{\Delta x_k \cdot \|\nabla \mathbf{f}(x_k)\|}_{} \\ &= \sum_k \ \nabla \mathbf{f}(x_k') \ \cdot \ \Delta x_k' \end{aligned} \tag{91}$$

(91) shows that normalization of column k of the sensitivity matrix to length 1 corresponds to scaling parameter x_k by the reciprocal norm of the column k.

An equalization of the sensitivity matrix is an additional measure to the initial scaling of performance features and parameters. The resulting improvement in the numerical condition can be interpreted as follows: Along with the equalization of the sensitivity values goes an equalization of the visibility of parameters and performance features within an optimization process. Optimization directions will be considered that otherwise would have been obscured by dominating numerical entries in the sensitivity matrix.

The formulas in this book are given for unscaled variables for a better illustration of the technical tasks. Scaled variables have to be used in an implementation.

4.4 Nominal Design

Nominal design naturally optimizes the performance features without considering parameter tolerances. In the following, the multiple-objective approach and the single-objective approach to performance optimization will be described.

Note that the description holds as well for tolerance design objectives. For that reason, the description is not put into subsections of this section, but into own sections.

4.5 Multiple-Objective Optimization

While a unique solution has been calculated explicitly for the RC circuit in Chapter 2, nominal design in general involves a multiple-objective optimization problem or multiple-criteria optimization problem respectively [35, 34, 41, 73, 66, 86], which can be formulated as:

$$\min_{\mathbf{x}_d} \ \mathbf{f}(\mathbf{x}_d) \quad \text{subject to} \quad \mathbf{c}(\mathbf{x}_d) \geq \mathbf{0} \tag{92}$$

In (92), the vector inequality is defined as in (12). (92) is formulated as a minimization task without loss of generality. Performance features that have to be maximized are included by $\max f \ = \ -\min -f$.

The constraints $\mathbf{c}(\mathbf{x}_d) \geq \mathbf{0}$ basically describe technological and structural requirements concerning DC properties of transistors that have to be fulfilled for a proper function and robustness [58]. Usually these constraints determine the achievable performance feature values of a given circuit structure.

Nominal design according to (92) is concerned with tuning the design parameters $\mathbf{x}_d$ in order to obtain an optimal performance. Doing so, it assumes that the statistical parameters $\mathbf{x}_d$ and the range parameters $\mathbf{x}_r$ have certain values. These values can be selected such that they represent worst-case conditions in statistical and range parameters.

If in addition, a performance specification according to (70) is given, then the multiple-objective minimization problem is formulated as:

$$\min_{\mathbf{x}_d} \ \mathbf{f}_I(\mathbf{x}_d) \quad \text{subject to} \quad \left\{ \begin{array}{l} \mathbf{f}_{II,L} \leq \mathbf{f}_{II}(\mathbf{x}_d) \leq \mathbf{f}_{II,U} \\ \mathbf{c}(\mathbf{x}_d) \geq \mathbf{0} \end{array} \right. \tag{93}$$

$\mathbf{f}_I, \mathbf{f}_{II}$ denote subsets of performance features. This corresponds to the practical situation, where a subset of performance features are optimization objectives while others are kept as constraints concerning their performance-specification features. This happens for instance if two or three performance features shall be compared visually or if computational costs shall be kept low.

The simultaneous minimization of several performance features as in (92) and (93) requires a compromise, i.e. a "trade-off," between competing optimization objectives. Typical examples of such trade-off situations are speed vs. power or gain vs. bandwidth. The optimality of one performance feature in a multiple-objective optimization problem can only be evaluated in connection with other performance features. This leads to the concept of Pareto optimality.

Definition. *A performance feature is called* **Pareto optimal** *if it can only be improved at the prize of deteriorating another performance feature.*

4.5.1 Smaller or Greater Performance Vectors

Pareto optimality is based on a relation of "less than" and "greater than" between performance vectors:

$$\mathbf{f} > \mathbf{f}^* \Leftrightarrow \ \mathbf{f} \geq \mathbf{f}^* \wedge \ \mathbf{f} \neq \mathbf{f}^* \ \Leftrightarrow \ \bigvee_i f_i \ \geq f_i^* \ \wedge \ \ \exists_i \ f_i \neq f_i^* \tag{94}$$

$$\mathbf{f} < \mathbf{f}^* \Leftrightarrow \ \mathbf{f} \leq \mathbf{f}^* \wedge \ \mathbf{f} \neq \mathbf{f}^* \ \Leftrightarrow \ \bigvee_i f_i \ \leq f_i^* \ \wedge \ \ \exists_i \ f_i \neq f_i^* \tag{95}$$

(94) and (95) define that a vector is less (greater) than another vector if all its components are less (greater) or equal and if it differs from the other vector.

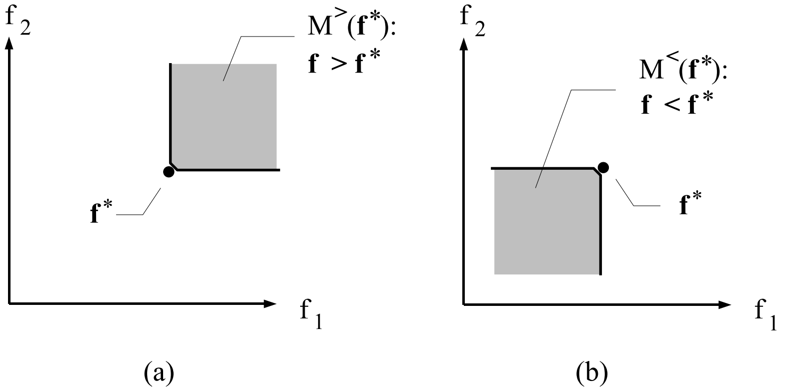


Figure 24. *(a)* Set $M^{>}(\mathbf{f}^*)$ of all performance vectors that are greater than $\mathbf{f}^*$, i.e. inferior to $\mathbf{f}^*$ with regard to multiple-criteria minimization according to (92) and (93). *(b)* Set $M^{<}(\mathbf{f}^*)$ of all performance vectors that are less than $\mathbf{f}^*$, i.e. superior to $\mathbf{f}^*$ with regard to multiple-criteria minimization.

This equivalently means that a vector is less (greater) than another vector if at least one of its components is less (greater) than the corresponding component of the other vector. In Figure 24(a), the shaded area indicates all performance vectors that are greater than a given performance vector $\mathbf{f}^*$ for a two-dimensional performance space.

These form the set $M^{>}(\mathbf{f}^*)$:

$$M^{>}(\mathbf{f}^*) = \{\mathbf{f} \mid \mathbf{f} > \mathbf{f}^*\} \tag{96}$$

The solid line at the border of the shaded region is meant as a part of $M^{>}(\mathbf{f}^*)$. The small kink of the solid line at $\mathbf{f}^*$ indicates that $\mathbf{f}^*$ is not part of $M^{>}(\mathbf{f}^*)$ according to the definition in (94). The solid line represents those performance vectors for which one performance feature is equal to the corresponding component of $\mathbf{f}^*$ and the other one is greater. In the shaded area in Figure 24(a), both performance features have values greater than the corresponding star values. With respect to a multiple-objective minimization problem, performance vectors in $M^{>}(\mathbf{f}^*)$ are inferior to $\mathbf{f}^*$. Figure 24(b) illustrates the analogous situation for all performance vectors that are less than, i.e. superior to, a given performance vector $\mathbf{f}^*$ for a two-dimensional performance space. These form the set $M^{<}(\mathbf{f}^*)$:

$$M^{<}(\mathbf{f}^*) = \{\mathbf{f} \mid \mathbf{f} < \mathbf{f}^*\} \tag{97}$$

From visual inspection of Figure 24(b), we can see that the border of the set of superior performance vectors $M^{<}(\mathbf{f}^*)$ to $\mathbf{f}^*$ corresponds to the level contour of the l_∞-norm of a vector, which is defined as the max operation on the absolute values of its components. The set of superior performance vectors $M^<(\mathbf{f}^*)$ to $\mathbf{f}^*$ can therefore be formulated based on the max-operation or the l_∞-norm, if the performance vectors are assumed to be scaled such that they only have positive values, $\mathbf{f} \in \mathbb{R}_+^{n_f}$:

$$\mathbf{f} \in \mathbb{R}_+^{n_f} : M^<(\mathbf{f}^*) = \left\{ \mathbf{f} \;\middle|\; \max_i \,(f_i/f_i^*) \leq 1 \;\wedge\; \mathbf{f} \neq \mathbf{f}^* \right\} \tag{98}$$

$$= \{\mathbf{f} \mid \| [\ldots f_i/f_i^* \ldots] \|_\infty \leq 1 \;\wedge\; \mathbf{f} \neq \mathbf{f}^* \} \tag{99}$$

Based on (97)–(99), Pareto optimality is defined by excluding the existence of any superior performance vector to a Pareto-optimal performance vector:

$$\mathbf{f}^* \text{ is Pareto optimal } \;\Leftrightarrow\; M^<(\mathbf{f}^*) = \{\,\} \tag{100}$$

4.5.2 Pareto Point

From (92) and (98)–(100) follows the formulation of a Pareto-optimal point (Pareto-efficient point, Pareto point) $\mathbf{f}_I^*$ as the performance vector with the minimum value of its maximum component among all feasible performance vectors:

$$\mathbf{f}^*(\mathbf{w}) \;\leftarrow\; \min_{\mathbf{x}_d} \underbrace{\max_i w_i \cdot (f_i(\mathbf{x}_d) - f_{RP,i})}_{\|\mathbf{f}(\mathbf{x}_d) - \mathbf{f}_{RP}\|_{\infty,w}} \;\text{ s.t. }\; \mathbf{c}(\mathbf{x}_d) \geq \mathbf{0} \tag{101}$$

$$w_i \geq 0 \,, \;\; \sum_i w_i = 1 \,, \;\; f_i - f_{RP,i} \geq 0$$

(101) shows that the min-max operation, or the l_∞-norm respectively, is an inherent part of the definition of a Pareto-optimal point.

If the reference point $\mathbf{f}_{RP} = [\, f_{RP,1} \ldots f_{RP,n_f} \,]^T$ is chosen such that the weights w_i refer to shifted performance feature values $f_i - f_{RP,i}$ that are always positive, a Pareto point $\mathbf{f}^*$ can be addressed from $\mathbf{f}_{RP}$ through a unique weight vector $\mathbf{w} = [\, w_1 \ldots w_{n_f} \,]^T$.

The individual minima of the performance features can be used to define the reference point:

$$f_{RP,i} \;\equiv\; \min_{\mathbf{x}_d} f_i(\mathbf{x}_d) \text{ s.t. } \mathbf{c}(\mathbf{x}_d) \geq \mathbf{0} \tag{102}$$

4.5.3 Pareto Front

In a multiple-objective optimization problem exists a set of Pareto points, which is called Pareto-optimal front or simply Pareto front PF. Figure 25(a) illustrates a Pareto front for two performance features. The gray region denotes

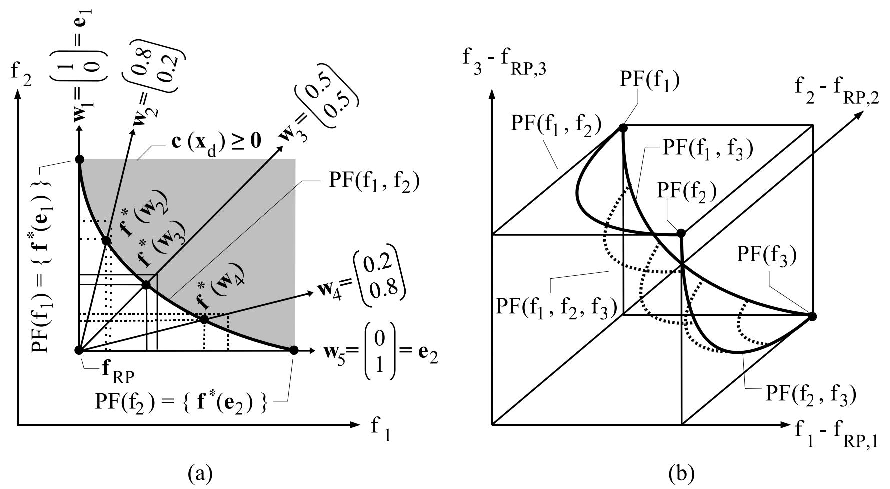


Figure 25. (a) Pareto front $PF(f_1, f_2)$ of a feasible performance region of two performance features. Different Pareto points addressed through different weight vectors from reference point $\mathbf{f}_{RP}$, which is determined by the individual minima of the performance features. *(b)* Pareto front $PF(f_1, f_2, f_3)$ of three performance features with boundary $\mathcal{B} = \{PF(f_1, f_2),\ PF(f_1, f_3),\ PF(f_2, f_3)\}$.

the set of all performance vectors that are achievable under consideration of the constraints of the optimization problem. The solid line at the left lower border of the region of achievable performance values represents the set of Pareto points, i.e. the Pareto front. Each point on the front is characterized in that no point superior to it exists, i.e. $M^{<}(\mathbf{f}^*)$ is empty. Three Pareto points are illustrated and the corresponding empty sets $M^{<}(\mathbf{f}^*)$ are indicated. We can as well see that each Pareto point represents the minimum weighted l_∞-norm with respect to the reference point $\mathbf{f}_{RP}$ with components according to (102) that is achievable for a given weight vector $\mathbf{w}$. Depending on the weight vector $\mathbf{w}$, a specific Pareto point on a ray from the reference point $\mathbf{f}_{RP}$ in the direction $\mathbf{w}$ is determined. The Pareto front PF and its boundary $\mathcal{B}$ can be defined as:

$$\begin{aligned} PF(\mathbf{f}) &= \{\mathbf{f}^*(\mathbf{w}) \mid w_i \geq 0\ ,\ \ \textstyle\sum_i w_i = 1\} \\ \mathcal{B}(PF(f_1, \ldots, f_{n_f})) &= \textstyle\bigcup_i PF(f_1, \ldots, f_{i-1}, f_{i+1} \cdots f_{n_f}) \end{aligned} \qquad (103)$$

A Pareto front for three performance features and its boundary is shown in Figure 25(b). Starting from the individual minima, the Pareto front can be formulated hierarchically by adding single performance features according to (103).

Pareto fronts may exhibit different shapes and may be discontinuous. Figure 26 shows three example shapes of Pareto fronts in relation to the convexity

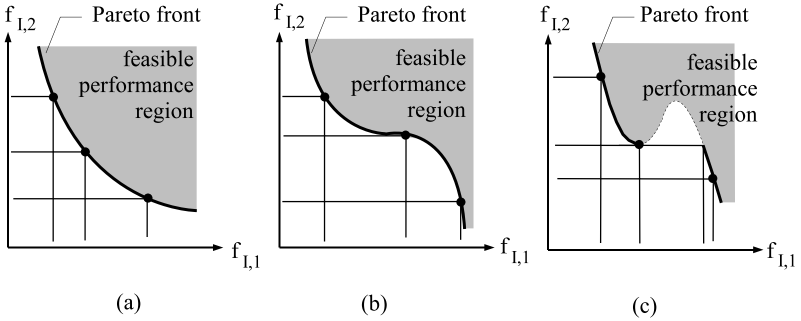


Figure 26. *(a)* Continuous Pareto front of a convex feasible performance region. $f^*_{I,2}$ is monotone in $f^*_{I,1}$. *(b)* Continuous Pareto front of a nonconvex feasible performance region. $f^*_{I,2}$ is monotone in $f^*_{I,1}$. *(c)* Discontinuous Pareto front of a nonconvex feasible performance region. $f^*_{I,2}$ is nonmonotone in $f^*_{I,1}$.

of the feasible performance region and the monotonicity of the explicit Pareto front function of one performance feature.

4.5.4 Pareto Optimization

The task of computing the Pareto front is called Pareto optimization. Usually the Pareto front is discretized by a number of Pareto points. The art of Pareto optimization is an efficient method to compute Pareto points that are evenly spread on the Pareto front. The computation of a Pareto point involves the solution of a single-objective optimization problem. Evolutionary algorithms as well as the deterministic Normal-Boundary-Intersection [34] and Goal-Attainment [48] approaches are suitable for Pareto optimization. A weighted sum approach with level contours (Figure 27(a)) that will end as tangents to the Pareto front is suitable for convex feasible performance regions only.

4.6 Single-Objective Optimization

We can distinguish two cases of single-objective optimization in analog design:

- Optimization of a selected single design target. E.g.:
  - As mentioned in the previous section, the individual minima of the performance features are very important in multiple-objective optimization. The individual minima are determined by optimizing a single performance feature while the other performance features are constraints or ignored.

  - The yield, i.e. the probability of satisfying the performance specification under given statistical parameter distributions and range-parameter tolerances, is an important single objective that has to be maximized.

- Optimization of a dedicated scalar function of several design targets. E.g.:
  - Multiple-objective optimization problems as in (92) and (93) require the solution of single-objective optimization problems as in (101) in order to calculate the Pareto front.
  - A set of performance features shall be optimized to meet the given performance specification with as much safety margin as possible. This is referred to as performance centering. Many kinds of scalar functions of several objectives are possible. However they introduce additional nonlinearities, as for instance a sum of exponential functions, and may deteriorate the numerical problem condition. The combination of multiple objectives into one scalar objective has to be defined very carefully.

A single-objective solution [19, 36, 91, 80, 45, 56, 61, 79] is approached with deterministic [115, 37, 81, 77, 28, 116, 83, 42, 105, 104, 16, 103, 27, 64, 62, 9, 92, 13, 12, 25, 23, 59, 7, 40, 74, 60, 2, 21] or statistical methods [33, 99, 26, 3, 46, 100, 110, 118, 96, 90, 93, 94, 102, 53, 111, 68, 69].

A practical problem formulation of single-objective analog optimization is:

$$\min_{\mathbf{x}_d} \; \|\mathbf{f}_I(\mathbf{x}_d) - \mathbf{f}_{I,target}\| \quad \text{subject to} \quad \begin{cases} \mathbf{f}_L \leq \mathbf{f}(\mathbf{x}_d) \leq \mathbf{f}_U \\ \mathbf{c}(\mathbf{x}_d) \geq \mathbf{0} \end{cases} \tag{104}$$

As mentioned, the constraints $\mathbf{c}(\mathbf{x}_d) \geq \mathbf{0}$ basically describe technological and structural requirements concerning DC properties of transistors that have to be fulfilled for a proper function and robustness [58]. The constraints $\mathbf{f}_L \leq \mathbf{f}(\mathbf{x}_d) \leq \mathbf{f}_U$ correspond to the performance specification. And a subset of performance features, $\mathbf{f}_I$, are selected as part of the optimization objective, which is a vector norm $\|.\|$ of their distances to target values, $\mathbf{f}_{I,target}$.

Single-objective analog optimization could of course be formulated as a minimum-norm problem without performance target values. But this would make the solution unnecessarily difficult. Because of the physical nature of the objectives, they cannot reach arbitrary values. Even achieving the individual minima of each performance feature simultaneously for all performance features is an utopian goal. Performance target values represent realistic goals for optimization that consider the practical conditions. They can be determined using input data and circuit knowledge and should be used in the sense of an appropriate scaling of performance features as described in Section 4.3 to ease the solution.

A concrete problem formulation now has to select the vector norm to be applied and find appropriate performance target values.

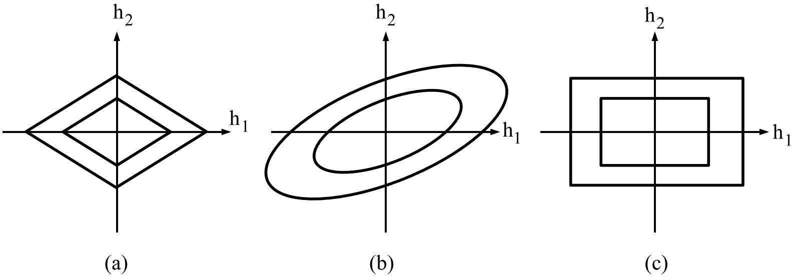


Figure 27. (a) Level contours of the weighted l_1-norm. *(b)* Level contours of the weighted l_2-norm. *(c)* Level contours of the weighted l_∞-norm.

4.6.1 Vector Norms

Several vector norms $\|.\|$ can be applied in (104):

- weighted l_1-norm

$$\|\mathbf{h}\|_{1,w} = |\mathbf{w}|^T \cdot |\mathbf{h}| = \sum_i |w_i| \cdot |h_i| \tag{105}$$

- weighted l_2-norm

$$\|\mathbf{h}\|_{2,w} = \sqrt{\mathbf{h}^T \cdot \mathbf{W} \cdot \mathbf{h}} \overset{\mathbf{W}=\text{diag}(w_i)}{=} \sqrt{\sum_i w_i^2 \cdot h_i^2} \tag{106}$$

- weighted l_∞-norm

$$\|\mathbf{h}\|_{\infty,w} = \max_i |w_i| \cdot |h_i| \tag{107}$$

Figure 27 illustrates the level contours of these three weighted norms.

The corners in their level contours illustrate that the l_1-norm and l_∞-norm are differentiable only with restrictions. The l_2-norm being differentiable makes it a favored norm for gradient-based numerical optimization.

Like the selected norm in (104), the selection of the performance target values $f_{I,target,i}$, $i = 1, \ldots, n_{fI}$ is very important.

4.6.2 Performance Targets

In the case of performance centering, the performance target values are defined using the performance-feature bounds. If a lower bound and an upper bound are specified for a performance feature at the same time, the target value is the center of the resulting interval:

$$\infty < f_{I,L,i} \leq f_{I,i} \leq f_{I,U,i} < \infty \ : \quad f_{I,target,i} = \frac{1}{2}\left(f_{I,L,i} + f_{I,U,i}\right) \quad (108)$$

This target value provides a maximum performance margin with regard to the respective performance-specification features.

If only an upper or only a lower bound is specified for a performance feature, a target value of this performance feature is hard to determine. It has to be determined based on experience and utilizing the sensitivity matrix. The performance target vector $\mathbf{f}_{I,target}$ should be updated during the optimization process.

4.7 Worst-Case Analysis and Optimization

In nominal design, the statistical and range parameters are considered at selected specific values. On the other hand, tolerance design considers a whole tolerance range of their values.

The task of worst-case analysis is to compute the worst-case performance that appears if the statistical parameters and the range parameters can take any value within their given tolerance regions.

Worst-case optimization hence is the task of minimizing the worst-case deviation of the performance from its nominal behavior.

Intuitively, the deviation of the worst-case from the nominal performance is proportional to the performance sensitivity. Then, worst-case optimization means a minimization of the performance sensitivity while taking care that the nominal performance does not drift away.

4.7.1 Worst-Case Analysis

Given a tolerance region of the statistical parameters T_s, a tolerance region of the range parameters T_r, and a nominal parameter vector for design parameters, statistical parameters and range parameters $\mathbf{x} = [\,\mathbf{x}_d^T \ \ \mathbf{x}_{s,0}^T \ \ \mathbf{x}_r^T\,]^T$, the worst-case analysis is formulated as:

$$f_i \geq f_{L,i} \ : \quad \min_{\mathbf{x}_s,\mathbf{x}_r} f_i(\mathbf{x}_s,\mathbf{x}_r) \ \text{ s.t. } \ \mathbf{x}_s \in T_s(Y_L)\ ,\ \mathbf{x}_r \in T_r \quad (109)$$

$$f_i \leq f_{U,i} \ : \quad \max_{\mathbf{x}_s,\mathbf{x}_r} f_i(\mathbf{x}_s,\mathbf{x}_r) \ \text{ s.t. } \ \mathbf{x}_s \in T_s(Y_L)\ ,\ \mathbf{x}_r \in T_r \quad (110)$$

$$i = 1, \ldots, n_f$$

If a lower bound $f_{L,i}$ is specified or required for a performance feature f_i, the worst-case represents the maximal deviation of the performance feature from its nominal value in negative direction. This value is obtained by computing the minimum value of the performance feature that occurs over the statistical and range parameters $\mathbf{x}_s$, $\mathbf{x}_r$ within their given tolerance regions T_s, T_r.

Vice versa, if an upper performance-feature bound $f_{U,i}$ is specified or required, the worst-case will be the maximal deviation of the performance feature from its nominal value in positive direction.

Even if no performance-feature bounds are given, it is often known if a worst-case of a performance feature happens in the direction of larger or of smaller values. The worst-case power for instance will be greater than the nominal power consumption, and a worst-case clock frequency will be less than the nominal clock frequency.

We may not be interested in a worst-case value for each performance feature. And if we are interested in the worst-case, then either a lower worst-case value or an upper worst-case value, or both a lower and upper worst-case value may be required, like the earliest and latest arrival times of a signal.

It follows that the number of worst-case performance values n_{WC} that we have to compute is

$$1 \leq n_{WC} \leq 2 \cdot n_f \tag{111}$$

The tolerance region of the range parameters T_r is given as an input and can have the forms in (8)-(11).

The tolerance region of the statistical parameters T_s depends on the chosen type of tolerance class and a required minimum yield Y_L. As the relation between yield and a tolerance region is ambiguous in the multivariate case, an a-priori determination of a specific tolerance region T_s for a given yield value Y_L has to be done heuristically (Section 6.2).

Note that the worst-case analysis (109), (110) in general requires the solution of an optimization problem.

The solution of the worst-case analysis problem (109), (110) yields worst-case parameter vectors $\mathbf{x}_{WL,i}$ and $\mathbf{x}_{WU,i}$ as well as the corresponding worst-case performance feature values $f_{WL,i}$ and $f_{WU,i}$.

The solution of (109) or (110) therefore implements the following mapping:

$$\begin{array}{lcl}
 & & \text{Worst-case parameter vectors} \\
 & & \mathbf{x}_{WL,i} = [\,\mathbf{x}_d^T \;\; \mathbf{x}_{s,WL,i}^T \;\; \mathbf{x}_{r,WL,i}^T\,]^T \\
 & & \mathbf{x}_{WU,i} = [\,\mathbf{x}_d^T \;\; \mathbf{x}_{s,WU,i}^T \;\; \mathbf{x}_{r,WU,i}^T\,]^T \\
\text{Tolerance region } \; T_s(Y_L) & & i = 1, \ldots, n_f \\
\text{Tolerance region } \; T_r & & \text{Worst-case performance feature values} \\
 & \mapsto & \\
\text{Nominal parameter vector} & & f_{WL,i} = f_i(\mathbf{x}_{WL,i}) \\
\mathbf{x} = [\,\mathbf{x}_d^T \;\; \mathbf{x}_{s,0}^T \;\; \mathbf{x}_r^T\,]^T & & \;\; \mathbf{f}_{WL,i} = \mathbf{f}(\mathbf{x}_{WL,i}) \\
 & & f_{WU,i} = f_i(\mathbf{x}_{WU,i}) \\
 & & \;\; \mathbf{f}_{WU,i} = \mathbf{f}(\mathbf{x}_{WU,i}) \\
 & & i = 1, \ldots, n_f
\end{array} \tag{112}$$

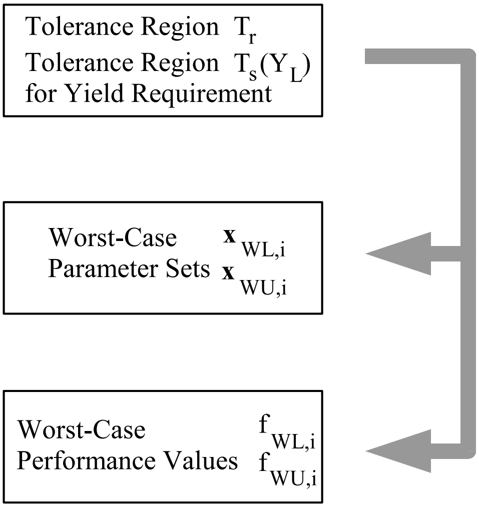

Tolerance Region T_r
Tolerance Region $T_s(Y_L)$
for Yield Requirement
Worst-Case $\mathbf{x}_{WL,i}$
Parameter Sets $\mathbf{x}_{WU,i}$
Worst-Case $f_{WL,i}$
Performance Values $f_{WU,i}$

Figure 28. Input and output of a worst-case analysis.

In each worst-case parameter vector $\mathbf{x}_{s,WL,i}$, $\mathbf{x}_{s,WU,i}$, where the performance feature f_i has its worst-case value $f_{WL,i}$, $f_{WU,i}$, the values of the other performance features add to the worst-case performance vectors $\mathbf{f}(\mathbf{x}_{WL,i})$, $\mathbf{f}(\mathbf{x}_{WU,i})$. A worst-case performance vector $\mathbf{f}(\mathbf{x}_{WL,i})$, $\mathbf{f}(\mathbf{x}_{WU,i})$ hence describes the situation of all performance features, if performance feature f_i is at its individual lower or upper worst-case. Figure 28 shows a graphical overview of the input and output of this mapping.

Figure 29 illustrates the input and output of a worst-case analysis in the spaces of design, statistical and range parameters and of performance features. For graphical reasons, each of these spaces is two-dimensional in this example. Figure 29(a) shows the input situation with nominal values for design parameters $\mathbf{x}_d$, statistical parameters $\mathbf{x}_{s,0}$, and range parameters $\mathbf{x}_r$, and with tolerance regions of the range parameters T_r and of the statistical parameters T_s. The tolerance regions have the typical shape of a box for the range parameters according to (8) and of an ellipsoid for the statistical parameters according to (10), (23) and (24). The tolerance region of statistical parameters has been determined to meet a certain yield requirement Y_L according to Section 6.2. Note that the design parameters do not change their values during worst-case analysis. A worst-case analysis computes lower and upper worst-case values for each of the two performance features. The results are illustrated in Figure 29(b) and consist of:

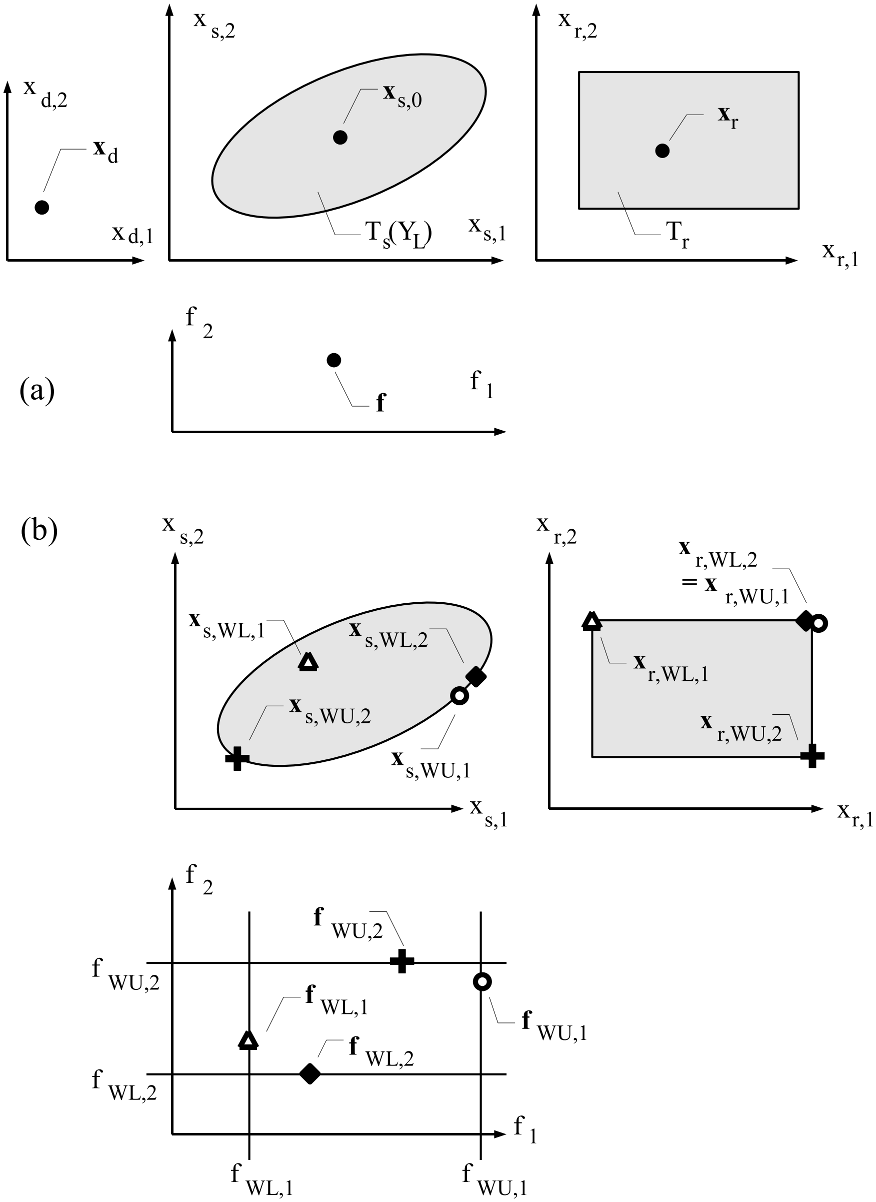


Figure 29. *(a)* Input of a worst-case analysis in the parameter and performance space: nominal parameter vector, tolerance regions. *(b)* Output of a worst-case analysis in the parameter and performance space: worst-case parameter vectors, worst-case performance values.

- four worst-case performance values $f_{WL,1}$, $f_{WU,1}$, $f_{WL,2}$ and $f_{WU,2}$,
- four worst-case parameter vectors $\mathbf{x}_{WL,i} = [\mathbf{x}_d^T\ \mathbf{x}_{s,WL,i}^T\ \mathbf{x}_{r,WL,i}^T]^T$, $\mathbf{x}_{WU,i} = [\mathbf{x}_d^T\ \mathbf{x}_{s,WU,i}^T\ \mathbf{x}_{r,WU,i}^T]^T$, $i = 1, 2$, which are given separately for the statistical parameters $\mathbf{x}_{s,WL,1}$, $\mathbf{x}_{s,WU,1}$, $\mathbf{x}_{s,WL,2}$ and $\mathbf{x}_{s,WU,2}$, and the range parameters $\mathbf{x}_{r,WL,1}$, $\mathbf{x}_{r,WU,1}$, $\mathbf{x}_{r,WL,2}$ and $\mathbf{x}_{r,WU,2}$, and
- the overall performance vectors at these parameter vectors $\mathbf{f}_{WL,1}$, $\mathbf{f}_{WU,1}$, $\mathbf{f}_{WL,2}$ and $\mathbf{f}_{WU,2}$.

Figure 29 illustrates the important characteristic of worst-case analysis that for each lower or upper direction of each performance feature, an individual worst-case parameter vector exists.

It may happen that worst-case parameter vectors are equal, illustrated by $\mathbf{x}_{r,WL,2}$ and $\mathbf{x}_{r,WU,1}$, or close to each other, illustrated by $\mathbf{x}_{s,WU,1}$, $\mathbf{x}_{s,WL,2}$ in Figure 29(b). An analysis of a clustering of worst-case parameter vectors for certain classes of performance features or certain classes of analog circuits may lead to a reduction in the required number of worst-case parameter vectors.

It may also happen that a worst-case parameter vector is not on the border of a tolerance region but inside the tolerance region, as illustrated by $\mathbf{x}_{s,WL,1}$ in Figure 29(b). If a performance function is unimodal over the parameters, than it has exactly one maximum or minimum. If this maximum or minimum is inside the parameter tolerance region, the corresponding upper or lower worst-case performance will be taken at the corresponding parameter values inside the tolerance region. If this maximum or minimum is outside of the parameter tolerance region, the maximum or minimum performance value will be on the border of the tolerance region.

Worst-case parameter vectors are very popular in analog design in order to check if a circuit satisfies the performance specification under tolerance conditions. (109) or (110) and the explanations above show that the computation of worst-case parameter vectors requires the cooperation of process technology and circuit design. The availability of statistical parameter distributions alone is not sufficient to compute worst-case parameter vectors. In addition, a concrete circuit and concrete performance features have to be provided by analog design.

4.7.2 Worst-Case Optimization

The goal of worst-case optimization is the minimization of the worst-case performance deviations from the nominal values, which have been computed by a worst-case analysis:

$$\min_{\mathbf{x}_d} \left\{ \begin{array}{l} |f_i(\mathbf{x}_d) - f_{WL,i}(\mathbf{x}_d)| \\ |f_{WU,i}(\mathbf{x}_d) - f_i(\mathbf{x}_d)| \\ i = 1, \ldots, n_f \end{array} \right\} \ \text{s.t.} \ \left\{ \begin{array}{l} f_{L,i} \le f_i / f_{WL/U,i}(\mathbf{x}_d) \le f_{U,i} \\ i = 1, \ldots, n_f \\ \mathbf{c}(\mathbf{x}_d) \ge \mathbf{0} \end{array} \right. \quad (113)$$

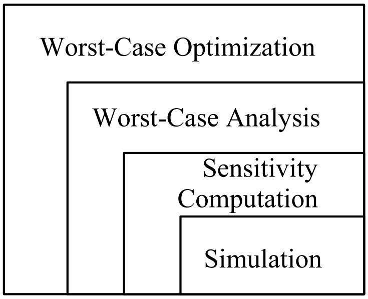


Figure 30. Nested loops within worst-case optimization.

During computation of the worst-case performance values over the statistical and range parameters, the design parameters have been kept at their nominal values.

Worst-case optimization utilizes that the nominal performance values and the worst-case performance values depend on the design parameters as well. By sizing the design parameter values $\mathbf{x}_d$, the amount of lower and upper worst-case deviations, $|f_i(\mathbf{x}_d) - f_{WL,i}(\mathbf{x}_d)|$ and $|f_{WU,i}(\mathbf{x}_d) - f_i(\mathbf{x}_d)|$, is minimized. At the same time, neither the worst-case nor the nominal performance values should violate the performance specification. This constraint can be relaxed to include only nominal performance values.

Worst-case optimization (113) is a multiple-objective optimization problem that has to be transformed into a suitable single-objective optimization problem. It targets worst-case performance values as optimization objectives, which in turn have been computed by means of an optimization problem. This problem partitioning necessitates a repeated worst-case analysis according to (109) or (110) within the iterative solution of the worst-case optimization process according to (113). Therefore worst-case optimization applies worst-case analysis "in a loop." It is thus a two-layered optimization process. Worst-case analysis in turn applies sensitivity computation for gradient calculation as well "in a loop." And if sensitivities are computed by finite differences, a sensitivity analysis in turn applies numerical simulation "in a loop." Figure 30 illustrates this situation of four nested loops.

Simplifications of this general approach result from the implementation of the loops in which worst-case optimization calls worst-case analysis and in which worst-case analysis calls sensitivity analysis. Simplifications are for instance:

- Call a worst-case analysis once for each iteration step of worst-case optimization.

- Call a worst-case analysis in every μth iteration step of worst-case optimization.

These are combined with:

- Call a sensitivity analysis once for each iteration step of a worst-case analysis.
- Call a sensitivity analysis once in a worst-case analysis.

A comparison of the worst-case optimization problem according to (113) and the nominal design problem according to (104) shows the similarity between these two design tasks. In both tasks, certain performance values (nominal and/or worst-case) are tuned within a given performance specification. As nominal design in practice will consider at least an approximation of the worst-case behavior based on a linear performance model, for instance based on sensitivities, the transition between nominal design and worst-case optimization is smooth. In order to save design time and computational cost, nominal design can already include assumptions about the worst-case behavior.

4.8 Yield Analysis, Yield Optimization/Design Centering

The yield is the percentage of manufactured circuits that satisfy the performance specification in face of statistical parameter variations and range-parameter tolerances.

Yield analysis denotes the estimation of the yield, yield optimization/design centering denotes the maximization of yield by tuning of design parameters.

4.8.1 Yield

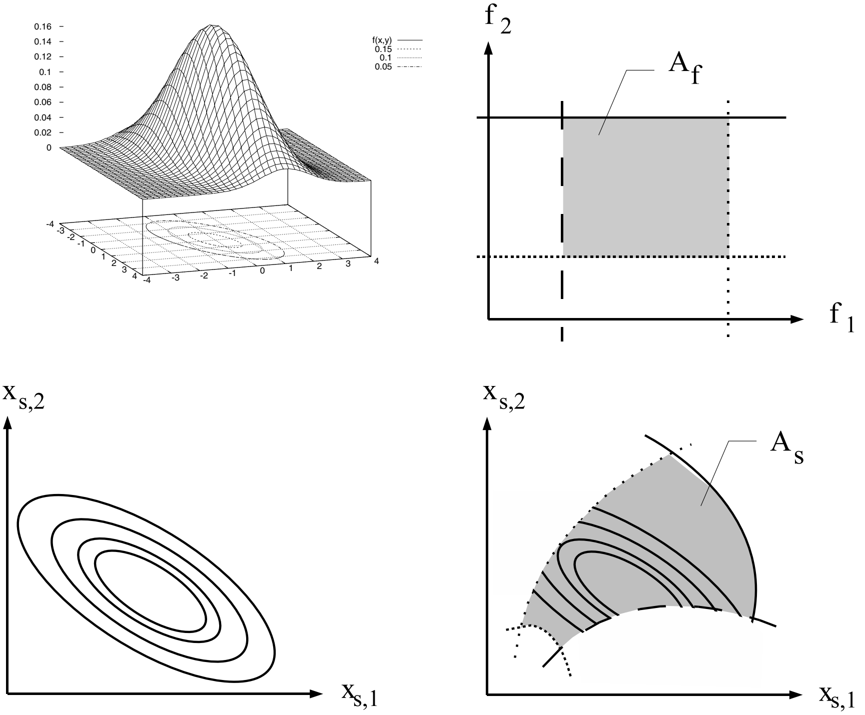
The yield Y can be defined as the probability that a manufactured circuit satisfies the performance specification under all operating conditions:

$$Y = \text{prob}\{ \underset{\mathbf{x}_r \in T_r}{\forall} \mathbf{f}_L \leq \mathbf{f}(\mathbf{x}_d, \mathbf{x}_s, \mathbf{x}_r) \leq \mathbf{f}_U \} \tag{114}$$

Depending on the perspective, the yield considered as a probability will be between 0 and 1, and considered as a percentage between 0% and 100%:

$$0 \leq Y \leq 1 \;\; \equiv \;\; 0\% \leq Y \leq 100\% \tag{115}$$

On the one hand, the yield is determined by the probability density function of the statistical parameters (13), (14). If no performance specification was given, then the yield would be 100% according to (15). Figure 31(a) shows a normal probability density function with its ellipsoidal equidensity contours, which is unaffected by any performance specification.



(a) (b)

Figure 31. *(a)* The volume under a probability density function of statistical parameters, which has ellipsoid equidensity contours, corresponds to 100% yield. *(b)* The performance specification defines the acceptance region A_f, i.e. the region of performance values of circuits that are in full working order. The yield is the portion of circuits in full working order. It is determined by the volume under the probability density function truncated by the corresponding parameter acceptance region A_s.

The volume under this probability density function is 1, which refers to 100% yield. On the other hand, the yield is determined by the performance specification (70), (67) and the range parameters' tolerance region (8)–(11). These lead to a yield loss because a certain percentage of the statistically varying parameters will violate the performance specification for some operating condition. Figure 31(b) illustrates how the ellipsoidal equidensity contours of the probability density function of statistical parameters are truncated by the parameter acceptance region A_s, which corresponds to the performance acceptance region A_f defined by the performance specification. Figure 17 earlier illustrated the remaining part of a normal probability density function if all parameter values that violate a performance specification are left out. The volume under this truncated probability density function refers to the probability that a circuit will satisfy the performance specification and is between 0 and 1, which refers to a yield value between 0% and 100%.

The yield Y is defined by integrating the probability density function over the acceptance region imposed by the performance specification. This can be done either in the statistical parameter space,

$$Y = \int \ldots \int_{\mathbf{x}_s \in A_s} \mathrm{pdf}(\mathbf{x}_s) \cdot d\mathbf{x}_s, \quad d\mathbf{x}_s = dx_{s,1} \cdot dx_{s,2} \cdot \ldots \cdot dx_{s,n_s} \tag{116}$$

or in the performance space:

$$Y = \int \ldots \int_{\mathbf{f} \in A_f} \mathrm{pdf}_f\,(\mathbf{f}) \cdot d\mathbf{f}, \qquad d\mathbf{f} = df_1 \cdot df_2 \cdot \ldots \cdot df_{n_f} \tag{117}$$

4.8.2 Acceptance Region Partitions

(116) and (117) illustrate that the yield formulation requires the formulation of both the probability density function and the acceptance region in either the performance space or the statistical parameter space. The acceptance region however is defined in the performance space, and the probability density function is defined in the statistical parameter space. Yield formulation (117) therefore requires an approximation of the probability density function of the performance features pdf_f, whereas yield formulation (116) requires an approximation of the acceptance region in the statistical parameter space A_s.

An approximation of pdf_f could be based on an expansion of the probability density function in dependence of statistical moments and an estimation of the statistical moments of the performance features.

An approximation of the acceptance region in the statistical parameter space starts from the formulation of the performance acceptance region A_f. A_f is partitioned into individual performance acceptance region partitions $A_{f,L,i}$, $A_{f,U,i}$, which refer to the individual performance-specification features:

$$A_f = \{\mathbf{f} \mid \mathbf{f}_L \le \mathbf{f} \le \mathbf{f}_U\} \tag{118}$$

$$A_{f,L,i} = \{\mathbf{f} \mid f_i \ge f_{L,i}\} \tag{119}$$

$$A_{f,U,i} = \{\mathbf{f} \mid f_i \le f_{U,i}\} \tag{120}$$

$$A_f = \bigcap_{i=1,\ldots,n_f} A_{f,L,i} \cap A_{f,U,i} \tag{121}$$

Correspondingly, the parameter acceptance region A_s, and individual parameter acceptance region partitions $A_{s,L,i}$, $A_{s,U,i}$ are defined:

$$A_s = \left\{ \mathbf{x}_s \left| \underset{\mathbf{x}_r \,\in\, T_r}{\forall} \; \mathbf{f}_L \leq \mathbf{f}(\mathbf{x}_s, \mathbf{x}_r) \leq \mathbf{f}_U \right. \right\} \tag{122}$$

$$A_{s,L,i} = \left\{ \mathbf{x}_s \left| \underset{\mathbf{x}_r \,\in\, T_r}{\forall} \; f_i(\mathbf{x}_s, \mathbf{x}_r) \geq f_{L,i} \right. \right\} \tag{123}$$

$$A_{s,U,i} = \left\{ \mathbf{x}_s \left| \underset{\mathbf{x}_r \,\in\, T_r}{\forall} \; f_i(\mathbf{x}_s, \mathbf{x}_r) \leq f_{U,i} \right. \right\} \tag{124}$$

$$A_s = \bigcap_{i=1,\ldots,n_f} A_{s,L,i} \cap A_{s,U,i} \tag{125}$$

The definitions of the parameter acceptance region and its partitions include that the respective performance-specification features have to be satisfied for all range-parameter vectors within their tolerance region. This more complicated formulation is due to the fact that the tolerance region of range parameters has the form of a performance specification but is an input to circuit simulation.

Figure 32 illustrates the parameter acceptance region and its partitioning according to performance-specification features. This partitioning can be of use for an approximation of A_s in (116). In Figure 32, the complementary “non-acceptance” region partitions are shown as well. The performance non-acceptance region $\bar{A}_f$ and its partitions are defined as:

$$\bar{A}_f = \left\{ \mathbf{f} \left| \underset{i}{\exists} \; f_i < f_{L,i} \vee f_i < f_{U,i} \right. \right\} \tag{126}$$

$$\bar{A}_{f,L,i} = \{ \mathbf{f} \mid f_i < f_{L,i} \} \tag{127}$$

$$\bar{A}_{f,U,i} = \{ \mathbf{f} \mid f_i > f_{U,i} \} \tag{128}$$

$$\bar{A}_f = \bigcup_{i=1,\ldots,n_f} \bar{A}_{f,L,i} \cup \bar{A}_{f,U,i} \tag{129}$$

The parameter non-acceptance region $\bar{A}_s$ and its partitions are defined as:

$$\bar{A}_s = \left\{ \mathbf{x}_s \left| \underset{\mathbf{x}_r \,\in\, T_r}{\exists} \; \underset{i}{\exists} \; \begin{array}{l} f_i(\mathbf{x}_s, \mathbf{x}_r) < f_{L,i} \vee \\ \vee f_i(\mathbf{x}_s, \mathbf{x}_r) > f_{U,i} \end{array} \right. \right\} \tag{130}$$

$$\bar{A}_{s,L,i} = \left\{ \mathbf{x}_s \left| \underset{\mathbf{x}_r \,\in\, T_r}{\exists} \; f_i(\mathbf{x}_s, \mathbf{x}_r) < f_{L,i} \right. \right\} \tag{131}$$

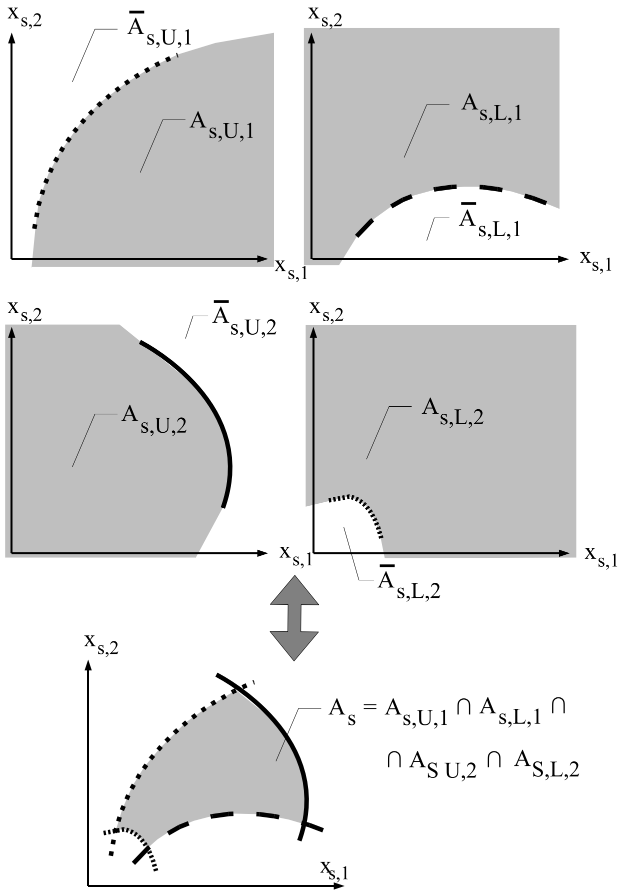


Figure 32. Parameter acceptance region A_s partitioned into parameter acceptance region partitions, $A_{s,L,1}$, $A_{s,U,1}$, $A_{s,L,2}$, $A_{s,U,2}$, for four performance-specification features, $f_1 \geq f_{L,1}$, $f_1 \leq f_{U,1}$, $f_2 \geq f_{L,2}$, $f_2 \leq f_{U,2}$. A_s results from the intersection of the parameter acceptance region partitions.

$$\bar{A}_{s,U,i} = \left\{ \mathbf{x}_s \,\middle|\, \underset{\mathbf{x}_r \in T_r}{\exists} f_i(\mathbf{x}_s, \mathbf{x}_r) > f_{U,i} \right\} \tag{132}$$

$$\bar{A}_s = \bigcup_{i=1,\dots,n_f} \bar{A}_{s,L,i} \cup \bar{A}_{s,U,i} \tag{133}$$

4.8.3 Yield Partitions

By introducing the acceptance functions $\delta(\mathbf{x}_s)$, $\delta_{L,i}(\mathbf{x}_s)$ and $\delta_{U,i}(\mathbf{x}_s)$,

$$\delta(\mathbf{x}_s) = \left\{ \begin{array}{ll} 1\,, & \mathbf{x}_s \in A_s \\ 0\,, & \mathbf{x}_s \in \bar{A}_s \end{array} \right\} = \left\{ \begin{array}{ll} 1\,, & \mathbf{f} \in A_f \\ 0\,, & \mathbf{f} \in \bar{A}_f \end{array} \right. \tag{134}$$

$$\delta_{L,i}(\mathbf{x}_s) = \left\{ \begin{array}{ll} 1\,, & \mathbf{x}_s \in A_{s,L,i} \\ 0\,, & \mathbf{x}_s \in \bar{A}_{s,L,i} \end{array} \right. \tag{135}$$

$$\delta_{U,i}(\mathbf{x}_s) = \left\{ \begin{array}{ll} 1\,, & \mathbf{x}_s \in A_{s,U,i} \\ 0\,, & \mathbf{x}_s \in \bar{A}_{s,U,i} \end{array} \right. \tag{136}$$

the yield can be formulated as the expectation value of the acceptance function according to Appendix A:

$$Y = \int_{-\infty}^{+\infty} \dots \int_{-\infty}^{+\infty} \delta(\mathbf{x}_s) \cdot \mathrm{pdf}(\mathbf{x}_s) \cdot d\mathbf{x}_s = \mathrm{E}\{\delta(\mathbf{x}_s)\} \tag{137}$$

$$Y_{L,i} = \mathrm{E}\{\delta_{L,i}(\mathbf{x}_s)\} \tag{138}$$

$$Y_{U,i} = \mathrm{E}\{\delta_{U,i}(\mathbf{x}_s)\} \tag{139}$$

Y represents the overall yield, $Y_{L,i}$ and $Y_{U,i}$ represent the yield partitions of the respective performance-specification features. An estimation of the yield according to (137)–(139) is based on statistical estimators according to Appendix B.

The yield partitions allow a ranking of the individual performance-specification features concerning their impact on the circuit robustness. Figure 32 illustrates that each added performance-specification feature usually leads to another yield loss. The smallest specification-feature yield value hence is an upper bound for the overall yield Y.

4.8.4 Yield Analysis

A yield analysis according to (137)–(139) calculates the yield and yield partition values for given performance-specification features, a given tolerance

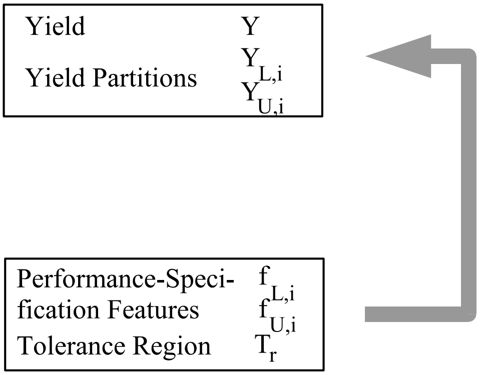

Yield Y
Yield Partitions $Y_{L,i}$ $Y_{U,i}$
Performance-Specification Features $f_{L,i}$ $f_{U,i}$
Tolerance Region T_r

Figure 33. Input and output of a yield analysis.

region of range parameters and a given nominal design:

$$\begin{array}{lcl} \text{Performance-specification features} & & \\ f_{L,i}\,,\ f_{U,i}\,,\ i = 1, \ldots, n_f & & \text{Yield } Y \\ \text{Tolerance region } T_r & \mapsto & \text{Yield Partitions} \\ \text{Nominal design} & & Y_{L,i}\,,\ Y_{U,i}\,,\ i = 1, \ldots, n_f \\ \mathbf{x} = [\, \mathbf{x}_d^T \ \ \mathbf{x}_{s,0}^T \ \ \mathbf{x}_r^T \,]^T & & \end{array} \tag{140}$$

This mapping is illustrated in Figure 33.

The yield analysis can be related to a worst-case analysis, which was illustrated in Figure 28. The worst-case performance values from a worst-case analysis become upper or lower performance-feature bounds as input of a yield analysis. The yield value that is obtained as an output of yield analysis becomes an input yield requirement of the worst-case analysis where it is transformed to a tolerance region of statistical parameters according to Section 6.2.

4.8.5 Yield Optimization/Design Centering

The goal of yield optimization/design centering is the maximization of the yield values computed by a yield analysis. According to (137)–(139) yield optimization/design centering can be formulated either as a single-objective optimization problem taking the yield,

$$\max_{\mathbf{x}_d} Y(\mathbf{x}_d) \ \text{ s.t. } \ \mathbf{c}(\mathbf{x}_d) \geq \mathbf{0} \tag{141}$$

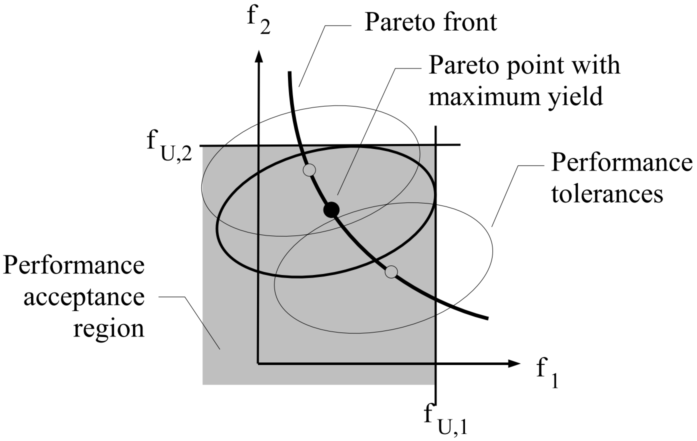


Figure 34. Yield optimization/design centering determines a selected point of Pareto front of performance features.

or as a multiple-objective optimization problem taking the yield partitions:

$$\max_{\mathbf{x}_d} \left\{ \begin{array}{l} Y_{L,i}(\mathbf{x}_d) \\ Y_{U,i}(\mathbf{x}_d) \\ i = 1, \ldots, n_f \end{array} \right\} \text{ s.t. } \mathbf{c}(\mathbf{x}_d) \geq \mathbf{0} \tag{142}$$

Interestingly, nominal design (93) and worst-case optimization (113) inherently are multiple-objective optimization problems, whereas yield optimization/design centering (141) inherently is a single-objective optimization problem. Apparently, it is difficult to combine various performance features into a single optimization objective. The solution of this task requires a mathematical approach like vector norms to combine multiple performance features into a single objective. The combination of parameter tolerances and the performance specification leads to an inherent single-objective analog design target, i.e. the yield. The solution of the yield optimization/design centering problem refers to a restricted part of the Pareto front of performance features, or even a single point on it, which is determined by the performance specification and the performance tolerances. Figure 34 illustrates an example of a Pareto front with three Pareto points and corresponding tolerance ellipsoids.

The performance tolerances have been assumed as equal for the three points, which is not the general case, but sufficient for illustration. As can be seen, the black point has maximum yield.

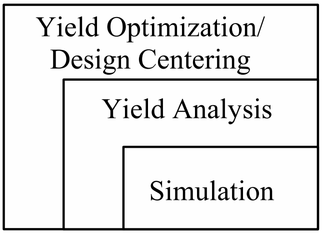


Figure 35. Nested loops within yield optimization/design centering.

Section 6.1.3 explains why a yield estimation is computationally expensive. Therefore, it may be advantageous to formulate yield optimization/design centering as a multiple-objective optimization problem (142). The partitioning into yield partition values may lead to an easier yield computation that pays off in terms of computational efficiency despite the approximation of the overall yield target.

An overview of the nested loops of yield optimization/design centering, which calls yield analysis in a loop, which in turn calls numerical simulation in a loop, is given in Figure 35. Similar to worst-case optimization, simplifications of this general approach result from the implementation of the loops in which yield optimization/design centering calls yield analysis and in which yield analysis calls simulation. Simplifications are for instance:

- Call a yield analysis once for each iteration step of yield optimization/design centering.
- Call a yield analysis in every kth iteration step of yield optimization/design centering.

These can be combined with the accuracy of the yield estimation in a yield analysis.

Figure 36 illustrates how yield optimization/design centering changes the situation in the statistical parameter space.

It shows the parameter acceptance region from Figure 32 plus level contours of a normal probability density function (23). Each level contour represents a constant value of the probability density function, which decreases with the size of the level contour. The parameter acceptance region depends on the values of design parameters that are disjunct from the statistical parameters. After yield optimization/design centering, the shape of the parameter acceptance region has changed in such a way that the volume of the truncated probability density function is at its maximum. This could look for instance as in Figure 36(b).

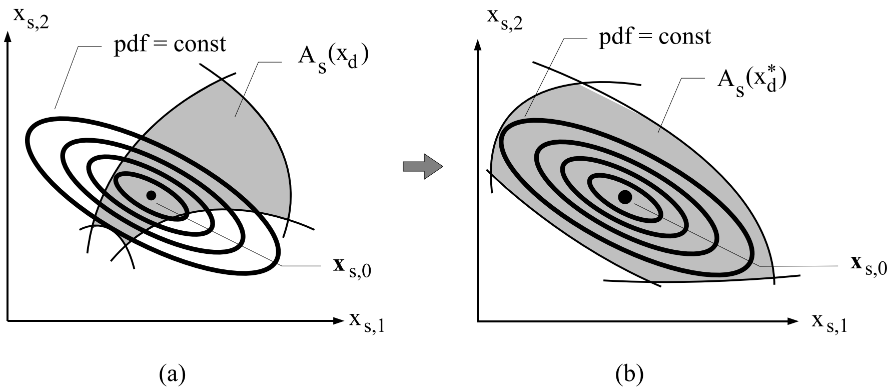


Figure 36. (a) Initial situation of yield optimization/design centering by tuning of design parameters $\mathbf{x}_d$ that are disjunct from statistical parameters. *(b)* After yield optimization/design centering by tuning of design parameters $\mathbf{x}_d$ that are disjunct from statistical parameters. The parameter acceptance region A_s depends on the values of design parameters $\mathbf{x}_d$. The equidensity contours of a normal probability density function are ellipsoids according to (24).

Note that maximum yield does not equally mean a maximum tolerance region inside the acceptance region.

The picture looks different if the design parameter space and the statistical parameter space are identical. In that case, the parameter acceptance region A_s will be constant. Yield optimization/design centering then happens through tuning of the statistical parameter distribution, which basically concerns the mean values, variances, correlations or higher-order moments. The first choice of yield optimization/design centering in this case is to tune the mean value $\mathbf{x}_{s,0}$:

$$\max_{\mathbf{x}_{s,0}} Y(\mathbf{x}_{s,0}) \text{ s.t. } \mathbf{c}(\mathbf{x}_{s,0}) \geq \mathbf{0} \qquad (143)$$

Figure 37(a) illustrates how yield optimization/design centering changes the situation in the statistical parameter space, if the nominal values of statistical parameters $\mathbf{x}_{s,0}$ are tuned. In Figure 36, the maximum yield had been achieved by appropriate changes in the acceptance region, now it is achieved by appropriate changes in the probability density function. Note that a combination of tuning of nominal values of design and of statistical parameters may occur as well.

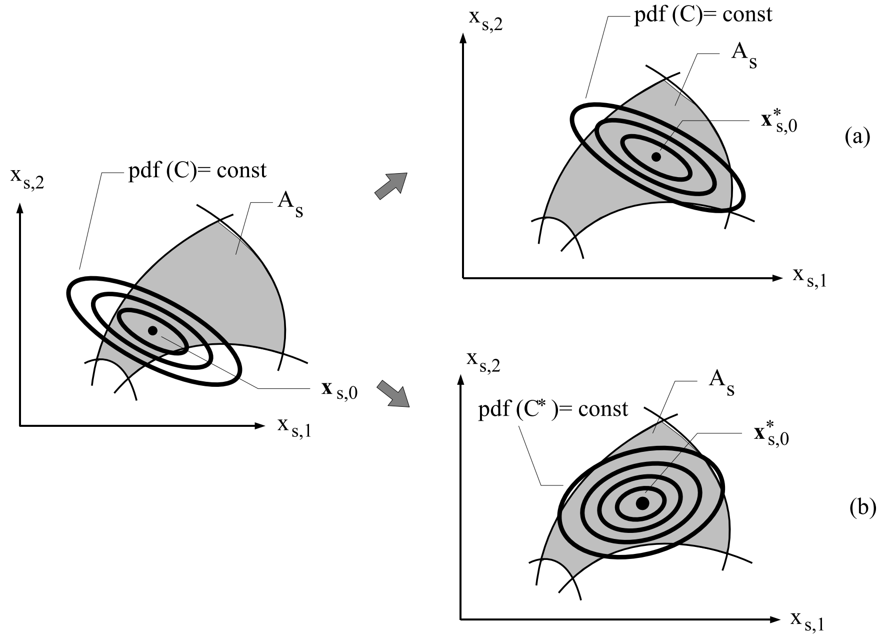


Figure 37. *(a)* Yield optimization/design centering by tuning of the mean values of statistical parameters $\mathbf{x}_{s,0}$. Parameter acceptance region A_s is constant. *(b)* Yield optimization/design centering by tuning of the mean values, variances and correlations of statistical parameters $\mathbf{x}_{s,0}$, $\mathbf{C}$ (tolerance assignment). Parameter acceptance region A_s is constant. Level contours of the normal probability density function (24) change their shape due to changes in the covariance matrix $\mathbf{C}$.

4.8.6 Tolerance Assignment

Figure 37(b) illustrates how yield optimization/design centering changes the situation in the statistical parameter space, if the nominal values, variances and correlations of statistical parameters are tuned. This type of yield optimization/design centering is called tolerance assignment:

$$\max_{\substack{\mathbf{x}_{s,0},\, \sigma_k,\, \varrho_{k,l} \\ k,l = 1,\ldots,n_{xs},\, k \neq l}} Y(\mathbf{x}_{s,0}, \mathbf{C}) \quad \text{s.t.} \quad \begin{cases} \mathbf{c}(\mathbf{x}_{s,0}) \geq \mathbf{0} \\ \det \mathbf{C} = \text{const} \neq 0 \end{cases} \tag{144}$$

Without the additional constraint concerning the covariance matrix $\mathbf{C}$ in (144), the trivial solution $\mathbf{C}^* = \mathbf{0}$ would be obtained. $\det \mathbf{C}$ describes the volume of a parallelepiped spanned by the columns or rows of $\mathbf{C}$. The volume $\det \mathbf{C}$ corresponds to the volume of a reference tolerance region, which is kept constant during the optimization to avoid the trivial solution.

From the definition of the parameter acceptance region and its partitions in (122)–(124) follows that the yield depends on the range-parameter bounds $x_{r,L,k}$, $x_{r,U,k}$, $k = 1, \ldots, n_{xr}$ as well.

Therefore, a yield sensitivity and yield improvement can be formulated with regard to the range-parameter bounds as another type of tolerance assignment.

Tolerance assignment plays a role in process tuning.

Tolerance assignment is also applied for the selection of discrete parameters with different tolerance intervals, like for instance resistors with $\pm 1\%$ tolerances or $\pm 10\%$ tolerances. Here the goal is to select the largest possible tolerance intervals without affecting the yield in order to save production costs.

Note that the yield also depends on the specified performance-feature bounds $f_{L,i}$, $f_{U,i}$, $i = 1, \ldots, n_f$, which can become a subject of yield optimization/design centering. Here the goal is to select the best possible performance specification that can be guaranteed with a certain yield.

4.8.7 Beyond 99.9% Yield

Figure 38 illustrates that the yield represents a very weak optimum for values above 99.9%, where further yield improvements are hardly achievable because the termination criteria of the optimization process will become active. If the design problem allows a yield above 99.9%, this property therefore results in a premature termination of the yield optimization/design centering process. In this case, the design cannot achieve its full robustness in terms of a parts-permillion yield loss and an increased robustness with respect to tightening of the performance specification and worsening of the parameter tolerances, as indicated in Figure 38.

To overcome the problem of premature termination, the parameter tolerances have to be controlled during the yield optimization process. Once a large enough yield value is reached, the tolerances are inflated. This scales the yield value down below 99% and avoids entering the region where the yield runs into saturation. In this way, the yield optimization/design centering process can continue to increase the scaled yield, and the true yield can go beyond 99.9%. This process is continued until no further improvement in the scaled yield is possible:

```
(1) $\max_{\mathbf{x}_d} Y(\mathbf{x}_d)$ s.t. $\mathbf{c}(\mathbf{x}_d) \geq \mathbf{0}$
(2) if $Y(\mathbf{x}_d^*) > 99\%$
(3)     $\mathbf{C} := a \cdot \mathbf{C}$ with $a > 1$                    (145)
(4)     goto (1)
(5) endif
```

In the case of tolerance assignment, a premature termination of the optimization

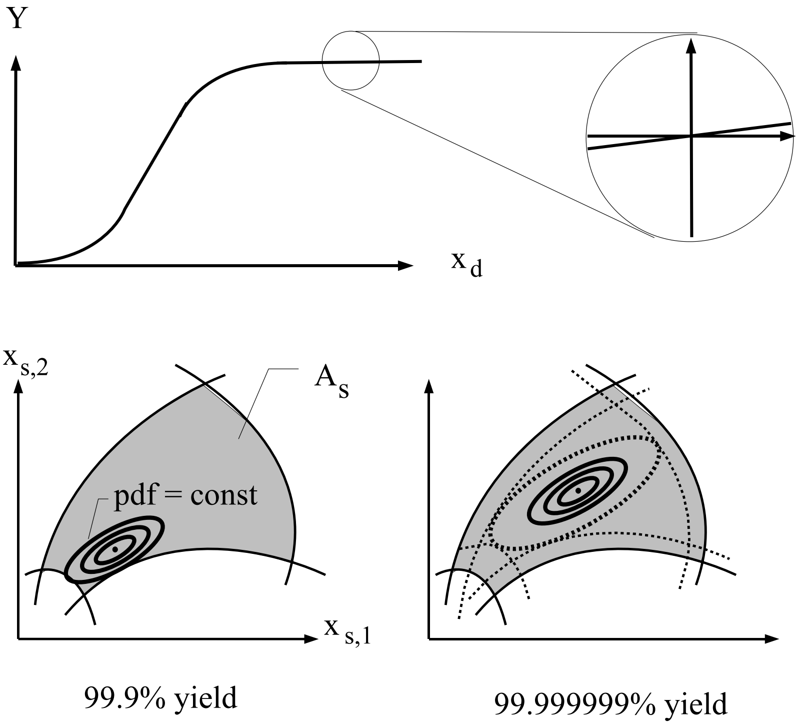


Figure 38. To go beyond 99.9% yield for maximum robustness, yield optimization/design centering requires specific measures.

process can be avoided by exchanging objective and constraint in (144):

$$\max_{\substack{\mathbf{x}_{s,0},\,\sigma_k,\,\varrho_{k,l} \\ k,l=1,\ldots,n_{xs},\,k\neq l}} \det \mathbf{C} \quad \text{s.t.} \quad \left\{ \begin{array}{l} \mathbf{c}(\mathbf{x}_{s,0}) \geq \mathbf{0} \\ Y(\mathbf{x}_{s,0}, \mathbf{C}) = \text{const} \equiv 50\% \end{array} \right. \tag{146}$$

In (146), the volume of the parallelepiped determined by the rows or columns of the covariance matrix is maximized while maintaining the yield at a constant value of for instance 50%. A yield value of 50% is advantageous because it is most sensitive to changes. This is illustrated in Figure 20, where the cumulative distribution function has its maximum slope if its value is 0.5. Solving problem (146) can be interpreted as inflating the set of parameter tolerance bodies as much as possible and adjusting their center without going below a certain yield value. It is intuitively clear that this will find a solution that can go beyond a yield of 99.9%.

A corresponding formulation of (146) for yield optimization/design centering, which only adjusts the nominal parameter vector and leaves the second-order moments unchanged, is:

$$\max_{\mathbf{x}_{s,0},\, a} \det(a \cdot \mathbf{C}) \;\text{ s.t. }\; \left\{ \begin{array}{l} \mathbf{c}(\mathbf{x}_{s,0}) \geq \mathbf{0} \\ Y(\mathbf{x}_{s,0}, a \cdot \mathbf{C}) = \text{const} \equiv 50\% \end{array} \right. \tag{147}$$

The solution of (147) will lead to a picture as in the lower right half of Figure 38.

A yield optimization/design centering is only complete if tries to go beyond yield values of 99.9% as described above. The geometric approach to yield optimization/design centering, described in Section 7.2, does not have the property of a weak optimum illustrated in Figure 38. It therefore leads to an optimum as illustrated on the lower right side of this figure without further ado.

Chapter 5

WORST-CASE ANALYSIS

In the following, three main types of worst-case analysis will be described. They differ in the type of tolerance region and in the modeling of the performance function and are suitable for different practical tasks. Table 6 gives an overview of the three approaches to worst-case analysis.

The classical worst-case analysis comes from a box-type tolerance region and a linear or linearized performance function. A box-type tolerance region results if tolerance intervals are defined for each individual parameter independently from the other parameters (8). Therefore the classical worst-case analysis is suitable for range parameters, for which the tolerance region is defined exactly in this way. The classical worst-case analysis is also suitable for uniformly distributed parameters, for which an interval of parameter values with equal probability density is defined, or if the type of distribution is not known.

The realistic worst-case analysis starts from an ellipsoid tolerance region and a linear or linearized performance function. An ellipsoid tolerance region results if the parameters are normally distributed (10), (23), (24). The realistic

Table 6. Worst-case analysis (WCA) types and characterization.

WCA type	Tolerance region	Performance function	Suitable for
Classical	Box	Linear	Uniform or unknown distribution Discrete or range parameters
Realistic	Ellipsoid	Linear	Normal distribution IC transistor parameters
General	Ellipsoid	Nonlinear	Normal distribution IC transistor parameters

worst-case analysis is therefore suitable for normally distributed parameters like transistor model parameters of integrated circuits.

We will illustrate that worst-case parameter vectors obtained by the classical worst-case analysis may correspond to an exaggerated robustness if applied to integrated circuits. This is mainly due to the missing consideration of the actual correlations. The realistic worst-case analysis considers the actual distribution of integrated circuits' parameters and achieves that the worst-case parameters represent more realistic yield values, that is the reason for its name.

Both classical and realistic worst-case analysis assume a linear function of the performance features in the parameters. Either the performance is linear in the parameters, or linearized performance models are used, for instance based on a sensitivity computation. A linearized model is generally not sufficient. This leads to the formulation of the general worst-case analysis, which starts from an ellipsoid tolerance region and a nonlinear performance in the parameters.

5.1 Classical Worst-Case Analysis

The classical worst-case analysis determines the worst-case value of one performance feature f for a given box tolerance region of range parameters $\mathbf{x}_r$,

$$\mathbf{x}_{r,L} \leq \mathbf{x}_r \leq \mathbf{x}_{r,U} \tag{148}$$

and for a given linear performance model,

$$\bar{f}(\mathbf{x}_r) = f_0 + \nabla f(\mathbf{x}_{r,0})^T \cdot (\mathbf{x}_r - \mathbf{x}_{r,0}) \tag{149}$$

based on (109) and (110):

$$f \geq f_L : \quad \min_{\mathbf{x}_r} + \nabla f(\mathbf{x}_{r,0})^T \cdot (\mathbf{x}_r - \mathbf{x}_{r,0}) \;\; \text{s.t.} \;\; \mathbf{x}_r \geq \mathbf{x}_{r,L}, \; \mathbf{x}_r \leq \mathbf{x}_{r,U} \tag{150}$$

$$f \leq f_U : \quad \min_{\mathbf{x}_r} - \nabla f(\mathbf{x}_{r,0})^T \cdot (\mathbf{x}_r - \mathbf{x}_{r,0}) \;\; \text{s.t.} \;\; \mathbf{x}_r \geq \mathbf{x}_{r,L}, \; \mathbf{x}_r \leq \mathbf{x}_{r,U} \tag{151}$$

In (150) and (151) the index i denoting the ith performance feature has been left out. (150) and (151) can be itemized concerning any performance feature and any type or subset of parameters.

Figure 39 illustrates the classical worst-case analysis problem in a two-dimensional parameter space for one performance feature. The gray area is the box tolerance region defined by a lower and upper bound of each parameter, $x_{r,L,1}$, $x_{r,U,1}$, $x_{r,L,2}$, $x_{r,U,2}$. The dotted lines are the level contours according to the gradient $\nabla f(\mathbf{x}_{r,0})$ of a linear performance model, which are equidistant planes. Each parameter vector on such a plane corresponds to a certain performance value. The plane through the nominal point $\mathbf{x}_{r,0}$ corresponds to the nominal performance value f_0. As the gradient points in the direction of steepest ascent, the upper and lower worst-case parameter vectors $\mathbf{x}_{r,WU}$ and

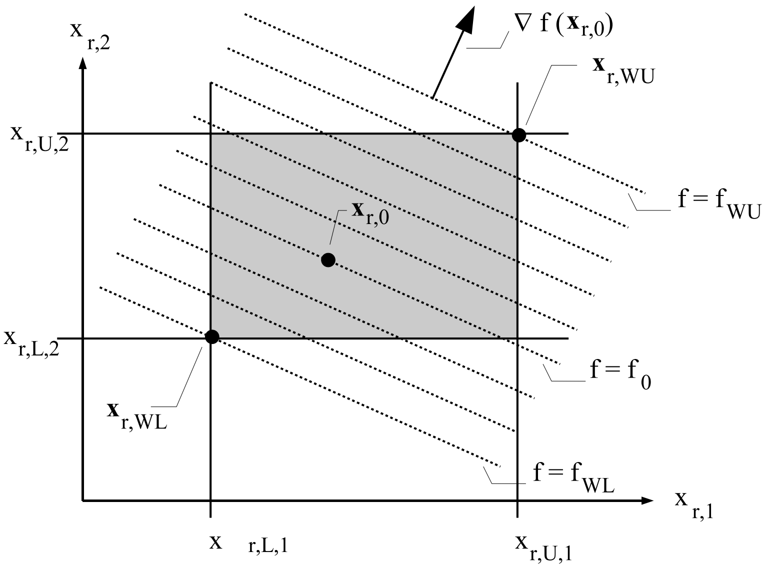


Figure 39. Classical worst-case analysis.

$\mathbf{x}_{r,WL}$ can readily be marked in. They correspond to the level contours of f that touch the tolerance region furthest away from the nominal level contour. This will usually happen in a corner of the tolerance box. These level contours through the worst-case parameter vectors represent the worst-case performance values f_{WU} and f_{WL}. The problem formulations (150) and (151) describe a special case of a linear programming problem that can be solved analytically.

Lower Worst-Case Performance. For an analytical solution, we first write the Lagrangian function of (150) according to Appendix C:

$$\mathcal{L}(\mathbf{x}_r, \boldsymbol{\lambda}_L, \boldsymbol{\lambda}_U) = \nabla f(\mathbf{x}_{r,0})^T \cdot (\mathbf{x}_r - \mathbf{x}_{r,0}) - \boldsymbol{\lambda}_L^T \cdot (\mathbf{x}_r - \mathbf{x}_{r,L}) - \boldsymbol{\lambda}_U^T \cdot (\mathbf{x}_{r,U} - \mathbf{x}_r) \tag{152}$$

Next, we write the first-order optimality condition of (152), which describes a solution of (150):

$$\nabla f(\mathbf{x}_{r,0}) - \boldsymbol{\lambda}_L + \boldsymbol{\lambda}_U = \mathbf{0} \tag{153}$$

$$\lambda_{L,k} \cdot (x_{r,k} - x_{r,L,k}) = 0\ , \quad k = 1, \ldots, n_{xr} \tag{154}$$

$$\lambda_{U,k} \cdot (x_{r,U,k} - x_{r,k}) = 0\ , \quad k = 1, \ldots, n_{xr} \tag{155}$$

(153) results from the condition that $\nabla\mathcal{L}(\mathbf{x}_r) = \mathbf{0}$ must hold for a stationary point of the optimization problem. (154) and (155) represent the complementarity condition of the optimization problem.

As either only the lower bound $x_{r,L,k}$ or the upper bound $x_{r,U,k}$ of a parameter $x_{r,k}$ can be active but not both of them, we obtain from (153) and the property that a Lagrange factor at the solution $\mathbf{x}_{r,WL}$ is positive:

$$\text{Either:} \quad \lambda_{WL,k} = +\nabla f(x_{r,0,k}) \; > \; 0 \tag{156}$$

$$\text{or:} \quad \lambda_{WU,k} = -\nabla f(x_{r,0,k}) \; > \; 0 \tag{157}$$

It depends on the sign of the gradient if (156) or (157) holds. Inserting either (156) in (154) or (157) in (155) leads to the formula of an element of a worst-case parameter vector for a worst-case lower performance value $x_{r,WL,k}$. Analogously an element of a worst-case parameter vector for a worst-case upper performance value $x_{r,WU,k}$ can be derived.

5.1.1 Classical Worst-Case Parameter Vectors

$\mathbf{x}_{r,WL/U} = [\ldots x_{r,WL/U,k} \cdots]^T$

$$x_{r,WL,k} = \begin{cases} x_{r,L,k}, & \nabla f(x_{r,0,k}) > 0 \\ x_{r,U,k}, & \nabla f(x_{r,0,k}) < 0 \\ \text{undefined}, & \nabla f(x_{r,0,k}) = 0 \end{cases} \tag{158}$$

$$x_{r,WU,k} = \begin{cases} x_{r,U,k}, & \nabla f(x_{r,0,k}) > 0 \\ x_{r,L,k}, & \nabla f(x_{r,0,k}) < 0 \\ \text{undefined}, & \nabla f(x_{r,0,k}) = 0 \end{cases} \tag{159}$$

Figure 40 illustrates the case when an element of the performance gradient is zero. The gradient with regard to parameter $x_{r,1}$ is zero. As the performance is insensitive with regard to $x_{r,1}$, the level lines are parallel to the $x_{r,1}$ coordinate axis. According to (158) and (159), any value of parameter $x_{r,1}$ is a solution of problems (150) and (151). The worst-case values $x_{r,WL,2}$, $x_{r,WU,2}$ of parameter $x_{r,2}$ are determined, but the worst-case values $x_{r,WL,1}$, $x_{r,WU,1}$ of parameter $x_{r,1}$ may therefore take any value in its defined tolerance interval, $x_{r,L,1} \leq x_{r,WL/U,1} \leq x_{r,U,1}$, as indicated as pieces of thick lines.

In practice, the independence of a performance feature value from a parameter, which results from a zero gradient, could be translated into a worst-case parameter that stays just at its nominal value.

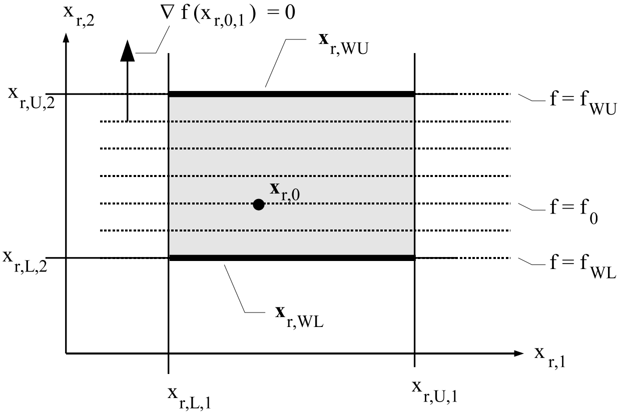


Figure 40. Classical worst-case analysis with undefined elements $x_{r,WL,1}$, $x_{r,WU,1}$, of worst-case parameter vectors $\mathbf{x}_{r,WL}$, $\mathbf{x}_{r,WU}$.

5.1.2 Classical Worst-Case Performance Values

The worst-case performance values result from insertion of the worst-case parameter vectors into the linear performance function (149):

$$f_{WL/U} = f_0 + \nabla f(\mathbf{x}_{r,0})^T \cdot (\mathbf{x}_{r,WL/U} - \mathbf{x}_{r,0}) \tag{160}$$

The index L/U means that the corresponding formula holds once for a lower performance-feature bound and once for an upper performance-feature bound.

5.1.3 Discrete Parameters

High-volume integrated circuit production will lead to a normal distribution of the parameters. If however discrete components from different productions are combined on a board, the distributions of these components will be quite different from each other. Figure 41 illustrates how the production test leads to truncated distributions for a component with different quality classes.

F refers to the clock frequency of a component. The test leads to classes ranging from very fast (F^{+++}) to slow (F^0), which are sold for prices ranging from high to low. The distributions for the different price categories will be truncated distributions.

If boards with discrete electronic components like resistors with selectable tolerance classes are composed, it becomes clear from Figure 41 that a $\pm 10\%$ resistor will most probably have a value very close to its $\pm 10\%$ value. Otherwise, this resistor would have been assigned to a narrower tolerance class and been sold at a higher price.

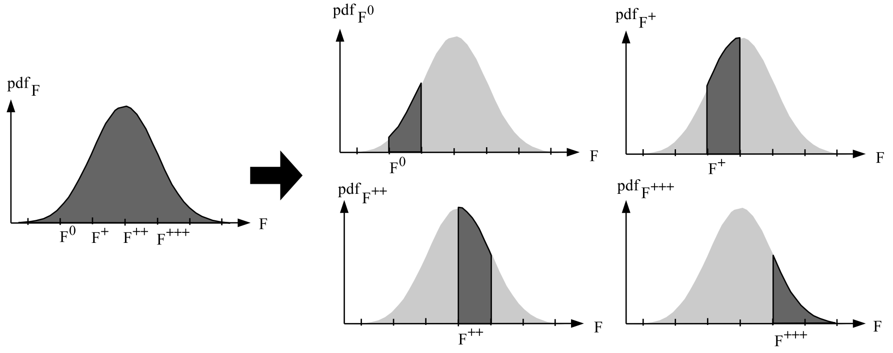


Figure 41. Normal probability density function of a manufactured component splits into truncated probability density functions after test according to different quality classes.

If a discrete system is made of such components, it is difficult to assume the type of distributions for the components. A classical worst-case analysis is an adequate choice for analyzing the impact of component tolerances in such a case.

5.1.4 Corner Worst Case

Classical worst-case parameter vectors applied in integrated circuit design are also referred to as corner worst case. For a specific performance feature, a set of relevant statistical parameters is selected. Each parameter is altered by a multiple of its standard deviation in either the positive or the negative direction of deteriorating performance. For the gate delay for instance, the resulting corner worst-case parameter vectors are the slow and fast worst-case parameter vectors. In order to consider more performance features, additional parameters have to be considered and more corner worst case parameter vectors result, like slow-slow, slow-fast, fast-fast.

5.2 Realistic Worst-Case Analysis

The realistic worst-case analysis assumes a joint distribution of parameters that are either normally distributed, or have been transformed into parameters that are normally distributed. The worst-case value of one performance feature f is then determined for a given ellipsoid tolerance region of parameters $\mathbf{x}_s$,

$$(\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \leq \beta_W^2 \qquad (161)$$

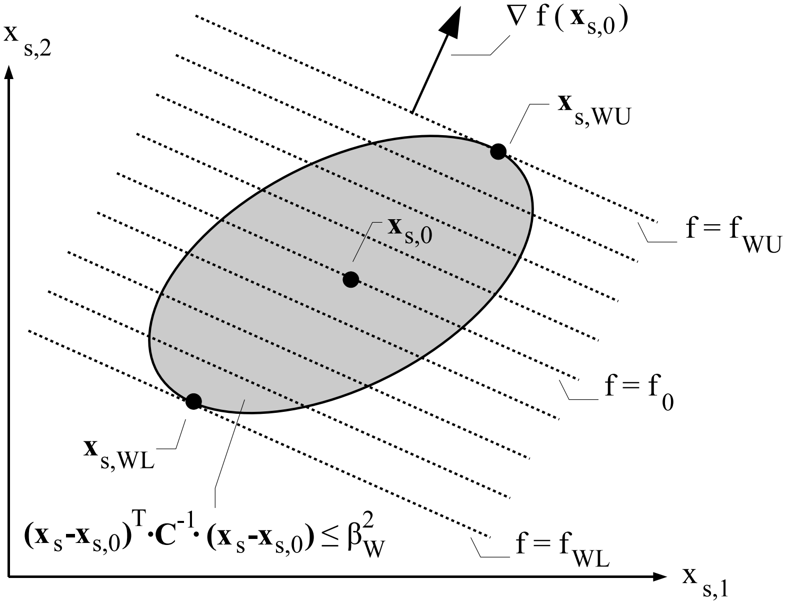

$x_{s,2}$
$\nabla f(\mathbf{x}_{s,0})$
$\mathbf{x}_{s,WU}$
$f = f_{WU}$
$\mathbf{x}_{s,0}$
$f = f_0$
$\mathbf{x}_{s,WL}$
$(\mathbf{x}_s-\mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s-\mathbf{x}_{s,0}) \leq \beta_W^2$
$f = f_{WL}$
$x_{s,1}$

Figure 42. Realistic worst-case analysis.

and for a given linear performance model,

$$\bar{f}(\mathbf{x}_s) = f_0 + \nabla f(\mathbf{x}_{s,0})^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \tag{162}$$

based on (109) and (110):

$$f \geq f_L : \quad \min_{\mathbf{x}_s} + \nabla f(\mathbf{x}_{s,0})^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \quad \text{s.t.} \ (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \leq \beta_W^2 \tag{163}$$

$$f \leq f_U : \quad \min_{\mathbf{x}_s} - \nabla f(\mathbf{x}_{s,0})^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \quad \text{s.t.} \ (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \leq \beta_W^2 \tag{164}$$

In (163) and (164) the index i denoting the ith performance feature has been left out. (163) and (164) can be itemized concerning any performance feature and any type or subset of parameters.

Figure 42 illustrates the realistic worst-case analysis problem in a two-dimensional parameter space for one performance feature. The gray area is the ellipsoid tolerance region defined by β_W, which is determined such that the worst-case represents a given yield requirement. Section 5.3 describes how this is done. The dotted lines are the level contours according to the gradient $\nabla f(\mathbf{x}_{s,0})$ of a linear performance model, which are equidistant planes. Each parameter vector on such a plane corresponds to a certain performance value. The plane through the nominal point $\mathbf{x}_{s,0}$ corresponds to the nominal performance value f_0.

As the gradient points in the direction of steepest ascent, the upper and lower worst-case parameter vectors $\mathbf{x}_{s,WU}$ and $\mathbf{x}_{s,WL}$ can readily be marked in Figure 42. They correspond to the level contours of f that touch the tolerance region furthest away from the nominal level contour. This happens somewhere on the border of the ellipsoid. The level contours through the worst-case parameter vectors represent the worst-case performance values f_{WU} and f_{WL}.

The problem formulations (163) and (164) describe a special programming problem with a linear objective function and a quadratic inequality constraint.

Lower Worst-Case Performance. An analytical solution can be derived based on the Lagrangian function of (163):

$$\begin{aligned} \mathcal{L}(\mathbf{x}_s, \lambda) \;=\;& \nabla f(\mathbf{x}_{s,0})^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \\ & -\lambda \cdot \left(\beta_W^2 - (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0})\right) \end{aligned} \tag{165}$$

The first-order optimality condition (Appendix C) of (165) describes a stationary point $\mathbf{x}_{s,WL}$, λ_{WL} of the Lagrangian function (165) and a solution of (163):

$$\nabla f(\mathbf{x}_{s,0}) + 2 \cdot \lambda_{WL} \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_{s,WL} - \mathbf{x}_{s,0}) = \mathbf{0} \tag{166}$$

$$(\mathbf{x}_{s,WL} - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_{s,WL} - \mathbf{x}_{s,0}) = \beta_W^2 \tag{167}$$

$$\lambda_{WL} > 0 \tag{168}$$

(166) results from the condition that $\nabla \mathcal{L}(\mathbf{x}_s) = \mathbf{0}$ must hold for a stationary point of the optimization problem. (167) and (168) consider that the inequality constraint has to be active in the solution because the objective function is linear.

The second-order optimality condition is satisfied because $\nabla^2 \mathcal{L}(\mathbf{x}_s) = 2 \cdot \lambda \cdot \mathbf{C}^{-1}$ is positive definite, as $\mathbf{C}^{-1}$ is positive definite and as $\lambda_{WL} > 0$.

From (166) we obtain:

$$\mathbf{x}_{s,WL} - \mathbf{x}_{s,0} = -\frac{1}{2\lambda_{WL}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0}) \tag{169}$$

Inserting (169) in (167) results in

$$\frac{1}{4\lambda_{WL}^2} \cdot \nabla f(\mathbf{x}_{s,0})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0}) = \beta_W^2 \tag{170}$$

Solving (170) for λ_{WL} and inserting the resulting equation for λ_{WL} in (169) leads to the analytical formulation of the worst-case parameter vector for a worst-case lower performance value $\mathbf{x}_{s,WL}$. Analogously a worst-case parameter vector for a worst-case upper performance value $\mathbf{x}_{s,WU}$ can be derived.

5.2.1 Realistic Worst-Case Parameter Vectors

$$\mathbf{x}_{s,WL} - \mathbf{x}_{s,0} = -\frac{\beta_W}{\sqrt{\nabla f(\mathbf{x}_{s,0})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0})}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0}) \qquad (171)$$

$$= -\frac{\beta_W}{\sigma_{\bar{f}}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0}) \qquad (172)$$

$$\mathbf{x}_{s,WU} - \mathbf{x}_{s,0} = +\frac{\beta_W}{\sqrt{\nabla f(\mathbf{x}_{s,0})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0})}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0}) \qquad (173)$$

$$= +\frac{\beta_W}{\sigma_{\bar{f}}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0}) \qquad (174)$$

As the performance feature $\bar{f}(\mathbf{x}_s)$ is linear in the normally distributed statistical parameters according to (162), its variance $\sigma_{\bar{f}}^2$ has been calculated using (A.13) and inserted in (172) and (174):

$$\sigma_{\bar{f}}^2 = \nabla f(\mathbf{x}_{s,0})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0}) \qquad (175)$$

Exercise. Apply (162) and (A.12) to prove (175).

5.2.2 Realistic Worst-Case Performance Values

The worst-case performance values result from insertion of the worst-case parameter vectors into the linear performance function (162), $f_{WL/U} = f_0 + \nabla f(\mathbf{x}_{s,0})^T \cdot (\mathbf{x}_{s,WL/U} - \mathbf{x}_{s,0})$:

$$f_{WL} = f_0 - \beta_W \cdot \sqrt{\nabla f(\mathbf{x}_{s,0})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0})} \qquad (176)$$

$$= f_0 - \beta_W \cdot \sigma_{\bar{f}} \qquad (177)$$

$$f_{WU} = f_0 + \beta_W \cdot \sqrt{\nabla f(\mathbf{x}_{s,0})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,0})} \qquad (178)$$

$$= f_0 + \beta_W \cdot \sigma_{\bar{f}} \qquad (179)$$

5.3 Yield/Worst-Case Distance – Linear Performance Feature

Assuming the performance feature f_i to be a linear function of the parameters (162), it is normally distributed. This is the result from the linear transformation of a normal distribution applying (A.13) in the same way as is done in step 3 in Section 3.8.

$$\bar{f}_i \sim \mathcal{N}\left(f_{0,i}, \sigma^2_{\bar{f}_i}\right) \tag{180}$$

The mean value of the probability density function is the nominal performance value in (162), the variance is given by (175).

From (177), (179) and (180), and applying (17), a unique relationship between the yield of a performance feature and the value of β_W, which characterizes the ellipsoid tolerance region, can be established:

$$\mathrm{pdf}_{\bar{f}_i}(\bar{f}_i) = \frac{1}{\sqrt{2\pi}\cdot\sigma_{\bar{f}_i}}\ \mathrm{e}^{-\frac{1}{2}\left(\frac{\bar{f}_i - f_{0,i}}{\sigma_{\bar{f}_i}}\right)^2} \tag{181}$$

$$\begin{aligned} Y_{U,i} &= \int_{-\infty}^{f_{WU,i}} \mathrm{pdf}_{\bar{f}_i}(\bar{f}_i)\cdot d\bar{f}_i = \int_{-\infty}^{\frac{f_{WU,i}-f_{0,i}}{\sigma_{\bar{f}_i}}} \frac{1}{\sqrt{2\pi}}\ \mathrm{e}^{-\frac{1}{2}t^2}\cdot dt \\ &= \int_{-\infty}^{\beta_W} \frac{1}{\sqrt{2\pi}}\ \mathrm{e}^{-\frac{1}{2}t^2}\cdot dt & (182) \\ &= Y_{L,i} & (183) \\ Y_i &= \int_{f_{WL,i}}^{f_{WU,i}} \mathrm{pdf}_{\bar{f}_i}(\bar{f}_i)\cdot d\bar{f}_i = \int_{-\beta_W}^{\beta_W} \frac{1}{\sqrt{2\pi}}\ \mathrm{e}^{-\frac{1}{2}t^2}\cdot dt & (184) \end{aligned}$$

$Y_{L,i}$, $Y_{U,i}$ are the yield partition values regarding the lower and upper worst-case performance values $f_{WL,i}$, $f_{WU,i}$ of the considered performance feature f_i. Y_i is the yield value regarding performance feature f_i with both its lower and upper worst-case value.

These equations open up the possibility of a technical interpretation of β_W as the measure that relates an ellipsoid tolerance region to a required yield.

We call β_W the worst-case distance between the nominal value and the worst-case value of a performance feature. According to (177) and (179) it is measured in the unit "performance variance": the worst-case performance value is β_W times of the performance variance away from the nominal performance value.

$\beta_W = 3$ therefore refers to a three-sigma safety margin (three-sigma design) of a performance feature, and $\beta_W = 6$ refers to a six-sigma safety margin (six-sigma design).

As a multiple of the performance variance it can immediately be translated into a yield value for one performance feature according to (182)-(184) and Table 5. Vice versa, a yield value can be translated into the size of an ellipsoid tolerance region in a multivariate parameter space of a realistic worst-case analysis according to (167). Section 5.5 will show that the meaning and interpretation of the worst-case distance will be valid for nonlinear performance functions as well.

5.4 General Worst-Case Analysis

The general worst-case analysis assumes a joint distribution of parameters, which is either normal or has been transformed into a normal distribution. The worst-case value of one performance feature f is then determined for a given ellipsoid tolerance region of parameters $\mathbf{x}_s$,

$$(\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \leq \beta_W^2 \tag{185}$$

and for a general nonlinear performance feature in the parameters $f(\mathbf{x}_s)$, based on (109) and (110):

$$f \geq f_L : \quad \min_{\mathbf{x}_s} +f(\mathbf{x}) \;\; \text{s.t.} \;\; (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \leq \beta_W^2 \tag{186}$$

$$f \leq f_U : \quad \min_{\mathbf{x}_s} -f(\mathbf{x}) \;\; \text{s.t.} \;\; (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \leq \beta_W^2 \tag{187}$$

In (186) and (187) the index i denoting the ith performance feature has been left out. (186) and (187) can be itemized concerning any performance feature and any type or subset of parameters.

Figure 43 illustrates the general worst-case analysis problem in a two-dimensional parameter space for one performance feature. The gray area is the ellipsoid tolerance region defined by β_W, which is determined such that the worst-case represents a given yield requirement. This is done according to (182)–(184). For a three-sigma design for instance, $\beta_W = 3$ would be chosen. Section 5.5 motivates why (182)–(184) are well-suited in the general nonlinear case.

The dotted lines are the level contours of the nonlinear performance feature. Each parameter vector on such a level contour corresponds to a certain performance value. The contour line through the nominal point $\mathbf{x}_{s,0}$ corresponds to the nominal performance value f_0. In this example, the performance is unimodal in the parameters and its minimum value is outside of the parameter tolerance ellipsoid.

The upper and lower worst-case parameter vectors $\mathbf{x}_{s,WU}$ and $\mathbf{x}_{s,WL}$ can readily be marked in. They correspond to the level contours of f that touch the tolerance region furthest away from the nominal level contour. This happens somewhere on the border of the ellipsoid in this case. These level contours through the worst-case parameter vectors represent the worst-case performance values f_{WU} and f_{WL}.

The property that performance level contour and tolerance region border touch each other in the worst-case parameter vector corresponds to a plane tangential both to the performance level contour and the tolerance region border. This tangent can be interpreted by a linearized performance model $\bar{f}$ in the worst-case parameter vector. The linearization is based on the gradient of the performance in the worst-case parameter vector.

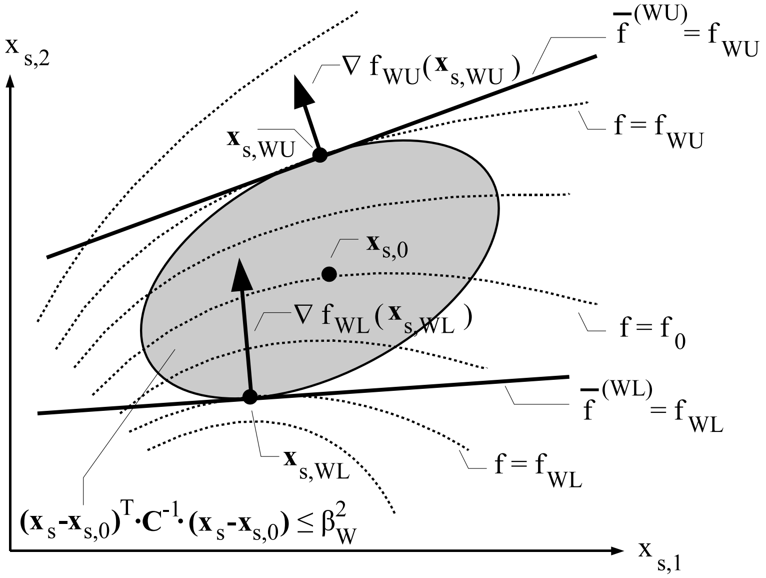

$x_{s,2}$
$x_{s,1}$
$\bar{f}^{(WU)} = f_{WU}$
$\nabla f_{WU}(\mathbf{x}_{s,WU})$
$f = f_{WU}$
$\mathbf{x}_{s,WU}$
$\mathbf{x}_{s,0}$
$\nabla f_{WL}(\mathbf{x}_{s,WL})$
$f = f_0$
$\bar{f}^{(WL)} = f_{WL}$
$\mathbf{x}_{s,WL}$
$f = f_{WL}$
$(\mathbf{x}_s-\mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s-\mathbf{x}_{s,0}) \leq \beta_W^2$

Figure 43. General worst-case analysis, in this case the solution is on the border of the tolerance region.

As the performance feature is nonlinear, we have a gradient $\nabla f(\mathbf{x}_{s,WU})$ at the upper worst-case parameter vector $\mathbf{x}_{s,WU}$ and a differing gradient $\nabla f(\mathbf{x}_{s,WL})$ at the lower worst-case parameter vector $\mathbf{x}_{s,WL}$ in this example. The linearization at the worst-case parameter vector depends on the respective gradient, worst-case performance value and worst-case parameter vector:

$$\bar{f}^{(WL/U)}(\mathbf{x}_s) \;=\; f_{WL/U} + \nabla f(\mathbf{x}_{s,WL/U})^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,WL/U}) \quad (188)$$

The level contour of such a linearized performance model through the respective worst-case parameter vector represents the corresponding worst-case performance value, $\bar{f}^{(WL/U)} = f_{WL/U}$.

In a realistic worst-case analysis, the performance function is linear in the parameters as illustrated in Figure 42. The solution of problems (163) and (164) is unique, and the worst-case parameter vectors will be on the border of the ellipsoid tolerance region.

In a general worst-case analysis, the solutions of problems (186) and (187) are not generally unique and the worst-case parameter vectors will be either on the border or inside of the ellipsoid tolerance region. It has been observed that usually unique solutions to problems (186) and (187) appear. An important case of multiple solutions are mismatch-sensitive performance functions with regard to local parameter changes, which are characterized by semidefinite second derivatives. The nominal performance is on a ridge and stays more or less constant in the direction of equal changes in two parameters and deteriorates in other directions. In such a case, very regularly concerning two local parameter variations two worst-case parameter vectors in “opposite” directions from the nominal parameter vector exist. Solution algorithms for problems (186) and (187) have to take care of such situations.

Problems (186) and (187) describe a special case of nonlinear programming with a nonlinear objective function and one quadratic inequality constraint. An analytical solution of problems (186) and (187) cannot be formulated. The solution is done by numerical optimization, for instance with a deterministic approach based on Sequential Quadratic Programming.

Lower Worst-Case Performance, Worst-Case Parameter Vector on the Border of the Tolerance Region. In the following, we will treat the computation of a lower worst-case performance value (186). The Lagrangian function of (186) is:

$$\mathcal{L}(\mathbf{x}_s, \lambda) = f(\mathbf{x}_s) - \lambda \cdot \left(\beta_W^2 - (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0})\right) \quad (189)$$

The first-order optimality condition (Appendix C) of (189) describes a stationary point $\mathbf{x}_{s,WL}$, λ_{WL} of the Lagrangian function (189) and a solution of (186):

$$\begin{aligned} \nabla f(\mathbf{x}_{s,WL}) + 2 \cdot \lambda_{WL} \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_{s,WL} - \mathbf{x}_{s,0}) &= \mathbf{0} \quad (190) \\ \lambda_{WL} \cdot \left(\beta_W^2 - (\mathbf{x}_{s,WL} - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_{s,WL} - \mathbf{x}_{s,0})\right) &= 0 \quad (191) \end{aligned}$$

(190) results from the condition that $\nabla \mathcal{L}(\mathbf{x}_s) = \mathbf{0}$ must hold for a stationary point of the optimization problem. (191) represents the complementarity condition of the optimization problem.

The second-order optimality condition holds because $\nabla^2 \mathcal{L}(\mathbf{x}_s) = 2 \cdot \lambda \cdot \mathbf{C}^{-1}$ is positive definite, as $\mathbf{C}^{-1}$ is positive definite and as $\lambda_{WL} \geq 0$.

We assume that the solution is on the border of the ellipsoid tolerance region. This happens for instance if the performance function $f(\mathbf{x}_s)$ is unimodal and if its maximum or minimum is outside of the parameters tolerance region. The constraint in (189) is therefore active:

$$\begin{aligned} \lambda_{WL} &> 0 \quad (192) \\ (\mathbf{x}_{s,WL} - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_{s,WL} - \mathbf{x}_{s,0}) &= \beta_W^2 \quad (193) \end{aligned}$$

Analogous equations for a worst-case parameter vector concerning a worst-case upper performance value $\mathbf{x}_{s,WU}$ can be derived.

(190), (192) and (193) have the same form as the first-order optimality condition of the realistic worst-case analysis (166)–(168). The general worst-case parameter vectors and general worst-case performance values therefore also have the same form as those of the realistic worst-case analysis. The only difference is in the gradient of the performance. One overall performance gradient appears in the realistic worst-case analysis, whereas an individual performance gradient at each worst-case parameter vector appears in the general worst-case analysis.

5.4.1 General Worst-Case Parameter Vectors

$$\mathbf{x}_{s,WL} - \mathbf{x}_{s,0} = -\frac{\beta_W}{\sqrt{\nabla f(\mathbf{x}_{s,WL})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL})}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL}) \quad (194)$$

$$= -\frac{\beta_W}{\sigma_{\bar{f}^{(WL)}}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL}) \quad (195)$$

$$\mathbf{x}_{s,WU} - \mathbf{x}_{s,0} = +\frac{\beta_W}{\sqrt{\nabla f(\mathbf{x}_{s,WU})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WU})}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WU}) \quad (196)$$

$$= +\frac{\beta_W}{\sigma_{\bar{f}^{(WU)}}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WU}) \quad (197)$$

In (195) and (197) the variance $\sigma^2_{\bar{f}^{(WL/U)}}$ of a performance function $\bar{f}^{(WL/U)}$ that is linear (188) in normally distributed parameters (25) has been inserted:

$$\sigma^2_{\bar{f}^{(WL/U)}} = \nabla f(\mathbf{x}_{s,WL/U})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U}) \quad (198)$$

5.4.2 General Worst-Case Performance Values

The worst-case performance values f_{WL} and f_{WU} result from the numerical solution of the optimization problems (186) and (187).

A formulation of the worst-case performance values that corresponds to the realistic worst-case analysis (176)–(179) can be derived by inserting the nominal parameter vector $\mathbf{x}_{s,0}$ into the linear performance functions (188):

$$\bar{f}^{(WL/U)}(\mathbf{x}_{s,0}) = \bar{f}_0^{(WL/U)} = f_{WL/U} + \nabla f(\mathbf{x}_{WL/U})^T \cdot (\mathbf{x}_{s,0} - \mathbf{x}_{WL/U})$$

$$\Leftrightarrow f_{WL/U} = \bar{f}_0^{(WL/U)} + \nabla f(\mathbf{x}_{WL/U})^T \cdot (\mathbf{x}_{WL/U} - \mathbf{x}_{s,0}) \quad (199)$$

Inserting (194)–(197) into (199) leads to:

$$f_{WL} = \bar{f}_0^{(WL)} - \beta_W \cdot \sqrt{\nabla f(\mathbf{x}_{s,WL})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL})} \quad (200)$$

$$= \bar{f}_0^{(WL)} - \beta_W \cdot \sigma_{\bar{f}^{(WL)}} \quad (201)$$

$$\begin{aligned} f_{WU} &= \bar{f}_0^{(WU)} + \beta_W \cdot \sqrt{\nabla f(\mathbf{x}_{s,WU})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WU})} \qquad (202) \\ &= \bar{f}_0^{(WU)} + \beta_W \cdot \sigma_{\bar{f}^{(WU)}} \qquad (203) \end{aligned}$$

(200)–(203) are equal to the corresponding equations (176)–(179) of the realistic worst-case analysis concerning the interpretation of the worst-case distance β_W according to Section 5.3. They differ in the definitions of the nominal performance value and the variance. In (200)–(203), the linearized models of the considered performance feature differ at the worst-case parameter vectors for the lower and upper worst-case performance value. This results in two different variance values $\sigma_{\bar{f}^{(WL/U)}}$ and in two nominal values of the performance feature $\bar{f}_0^{(WL/U)}$ that differ from the true nominal value f_0.

5.5 Yield/Worst-Case Distance – Nonlinear Performance Feature

With regard to the linear models (188) in the worst-case parameter vectors, these linearized performance features are normally distributed according to Section 3.8 and Appendix A.9:

$$\bar{f}_i^{(WL/U)} \sim \mathcal{N}\left(\bar{f}_{0,i}^{(WL/U)}, \sigma^2_{\bar{f}_i^{(WL/U)}}\right) \qquad (204)$$

The mean value of the probability density function is the value of the linearized performance function according to (199) in the nominal parameter vector, the variance is given by (198).

From (201), (203) and (204), and applying (17), a unique relationship between the yield of the performance feature linearized at the worst-case parameter vector and the value of β_W, which characterizes the ellipsoid tolerance region can be established:

$$\mathrm{pdf}_{\bar{f}_i^{(WL/U)}}(\bar{f}_i^{(WL/U)}) = \frac{1}{\sqrt{2\pi} \cdot \sigma_{\bar{f}_i^{(WL/U)}}} \, \mathrm{e}^{-\frac{1}{2}\left(\frac{\bar{f}_i^{(WL/U)} - \bar{f}_{0,i}^{(WL/U)}}{\sigma_{\bar{f}_i^{(WL/U)}}}\right)^2} \qquad (205)$$

$$\begin{aligned} \bar{Y}_{U,i} &= \int_{-\infty}^{f_{WU,i}} \mathrm{pdf}_{\bar{f}_i^{(WU)}}(\bar{f}_i^{(WU)}) \cdot d\bar{f}_i^{(WU)} \\ &= \int_{-\infty}^{\frac{f_{WU,i} - \bar{f}_{0,i}^{(WU)}}{\sigma_{\bar{f}_i^{(WU)}}}} \frac{1}{\sqrt{2\pi}} \, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt \\ &= \int_{-\infty}^{\beta_W} \frac{1}{\sqrt{2\pi}} \, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt \qquad (206) \end{aligned}$$

$$= \quad \bar{Y}_{L,i} \tag{207}$$

$\bar{Y}_{L,i}$ and $\bar{Y}_{U,i}$ are approximated yield partition values regarding the lower and upper worst-case performance value $f_{WL/U,i}$ of performance feature f_i.

This equation opens up the possibility of a technical interpretation of β_W as the measure that relates an ellipsoid tolerance region to an approximated yield.

We call β_W the worst-case distance between the nominal value and the worst-case value of a linearized performance feature according to (188). While the worst-case value of the real performance feature and the linearized performance feature are equal, the nominal value of the linearized performance feature differs from the real nominal value according to (199). According to (201) and (203), the worst-case distance is measured in the unit "variance of the linearized performance": the worst-case performance value is β_W times the variance away from the nominal linearized performance value. This variance is determined by (198).

$\beta_W = 3$ therefore refers to a three-sigma safety margin (three-sigma design) of a performance feature, and $\beta_W = 6$ refers to a six-sigma safety margin (six-sigma design).

As a multiple of a performance variance, it can immediately be translated into an approximated yield value for one performance feature according to (206) and (207) and Table 5. Vice versa, a yield value can be translated approximately into a value of β_W, which determines an ellipsoid tolerance region in a multivariate parameter space according to (191).

5.5.1 Yield Approximation Accuracy

An approximation error between the true yield partition value and the yield partition value according to (206), (207) results from approximating a nonlinear performance function $f_i(\mathbf{x}_s)$ by a linear model $\bar{f}_i^{(WL/U)}(\mathbf{x}_s)$.

This approximation error will be discussed in the following by investigating the parameter space. It will be assumed that the worst-case parameter vectors are on the border of the ellipsoid tolerance region.

According to the definition (116), (117), the yield can be formulated equivalently in the performance space and in the parameter space. With the formulations of acceptance regions in the performance space and in the parameter space in (118)–(125), the approximate yield partition $\bar{Y}_{L,i}$ with regard to a lower performance-feature bound $f_{WL,i}$ and the approximate yield partition $\bar{Y}_{U,i}$ with regard to an upper worst-case performance-feature value $f_{WU,i}$ can be formulated as:

$$\bar{Y}_{U,i} \quad = \quad \int \!\!\ldots\!\! \int_{\bar{f}_i^{(WU)}(\mathbf{x}_s) \leq f_{WU,i}} \mathrm{pdf}_N(\mathbf{x}_s) \cdot d\mathbf{x}_s \tag{208}$$

$$= \int \ldots \int_{\nabla f_i(\mathbf{x}_{s,WU,i})^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,WU,i}) \leq 0} \mathrm{pdf}_N(\mathbf{x}_s) \cdot d\mathbf{x}_s \qquad (209)$$

$$\bar{Y}_{L,i} = \int \ldots \int_{\bar{f}_i^{(WL)} \geq f_{WL,i}} \mathrm{pdf}_N(\mathbf{x}_s) \cdot d\mathbf{x}_s \qquad (210)$$

$$= \int \ldots \int_{\nabla f_i(\mathbf{x}_{s,WL,i})^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,WL,i}) \geq 0} \mathrm{pdf}_N(\mathbf{x}_s) \cdot d\mathbf{x}_s \qquad (211)$$

(188) has been applied to get from the yield definition in the performance space (208), (210) to the yield definition in the parameter space (209), (211).

pdf_N is the normal probability density function according to (23).

On the other hand, the true yield partition values, $Y_{L,i}$ with regard to a lower worst-case performance-feature value $f_{WL,i}$, and $Y_{U,i}$ with regard to an upper worst-case performance-feature value $f_{WU,i}$, can be formulated as:

$$Y_{U,i} = \int \ldots \int_{f_i(\mathbf{x}_s) \leq f_{WU,i}} \mathrm{pdf}_N(\mathbf{x}_s) \cdot d\mathbf{x}_s \qquad (212)$$

$$Y_{L,i} = \int \ldots \int_{f_i(\mathbf{x}_s) \geq f_{WL,i}} \mathrm{pdf}_N(\mathbf{x}_s) \cdot d\mathbf{x}_s \qquad (213)$$

(209), (211), (212) and (213) show that the error in the yield approximation results from the approximation of the parameter acceptance region concerning a performance-specification feature.

Figure 44 illustrates the situation for the example in Figure 43. In Figure 44(a), the upper worst-case has been picked. The ellipsoid tolerance region corresponds to the given value of β_W. The real level contour of all parameter vectors that lead to the worst-case performance value f_{WU} is shown by as a dotted curve. It represents the border of the acceptance region that determines the true yield partition value Y_U. This acceptance region is shaded in gray.

In addition, the level contour of all parameter vectors that lead to f_{WU} according to the linear performance model $\bar{f}^{(WU)}$ is shown as a solid line. It is tangential to the true level contour in the worst-case parameter vector $\mathbf{x}_{s,WU}$, because $\bar{f}^{(WU)}$ has been specifically established in $\mathbf{x}_{s,WU}$ by the general worst-case analysis. It represents the border of the approximate acceptance region that determines the approximate yield partition value $\bar{Y}_U$. This region is filled with a linen pattern.

The integration of the probability density function over the difference region between the linen-pattern-filled area of the approximate parameter acceptance region partition and the gray area of the true parameter acceptance region partition determines the error of the yield approximation. In Figure 44(a), the true yield will be overestimated. In Figure 44(b), the lower worst-case has

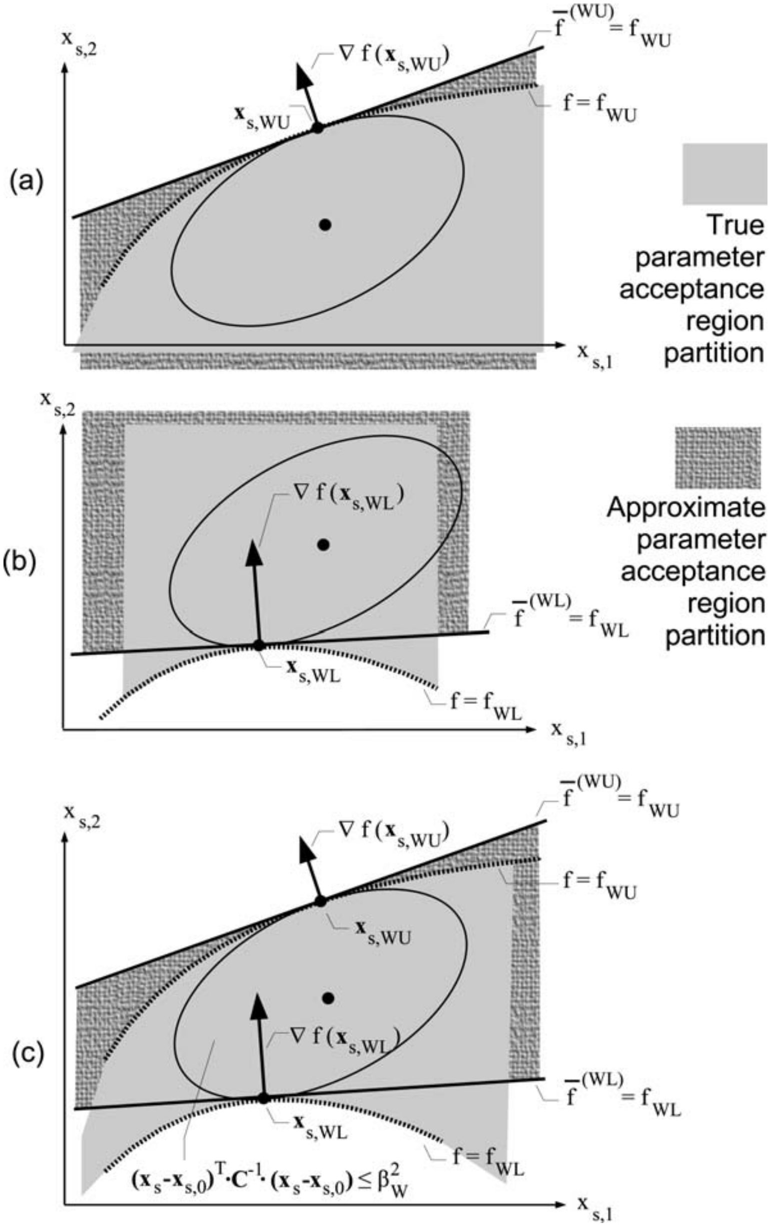

(a)
$x_{s,2}$
$x_{s,1}$
$\nabla f(\mathbf{x}_{s,WU})$
$\mathbf{x}_{s,WU}$
$\overline{f}^{(WU)} = f_{WU}$
$f = f_{WU}$
True parameter acceptance region partition
(b)
$x_{s,2}$
$x_{s,1}$
$\nabla f(\mathbf{x}_{s,WL})$
$\mathbf{x}_{s,WL}$
$\overline{f}^{(WL)} = f_{WL}$
$f = f_{WL}$
Approximate parameter acceptance region partition
(c)
$x_{s,2}$
$x_{s,1}$
$\nabla f(\mathbf{x}_{s,WU})$
$\mathbf{x}_{s,WU}$
$\overline{f}^{(WU)} = f_{WU}$
$f = f_{WU}$
$\nabla f(\mathbf{x}_{s,WL})$
$\mathbf{x}_{s,WL}$
$\overline{f}^{(WL)} = f_{WL}$
$f = f_{WL}$
$(\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \leq \beta_W^2$

Figure 44. *(a)* Comparison between true parameter acceptance region partitions (gray areas) and approximate parameter acceptance region partitions (linen-pattern-filled areas) of a general worst-case analysis of an upper worst-case performance value. *(b)* Lower worst-case performance value. *(c)* Lower and upper worst-case performance value.

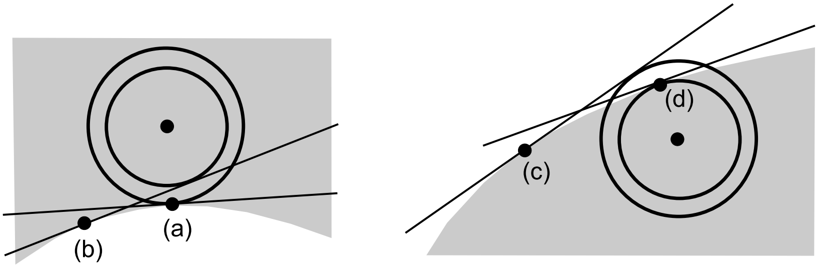


Figure 45. Duality principle in minimum norm problems. Shown are two acceptance regions (gray), the respective nominal points within acceptance regions and two points on the border of each acceptance region. Points (a) and (d) are the worst-case points.

been picked. Here, the true yield will be underestimated. Figure 44(c) shows the overall approximate acceptance region if the lower and upper worst-case parameter vectors and their linearizations are combined.

By inspection, the following statements concerning the approximation error result:

- Considering the decreasing values of the probability density function orthogonal to its ellipsoid level contours, the yield approximation error becomes more critical in regions that correspond to ellipsoids closer to the center. The chosen linearization is very well adapted to this property. It is exact in the worst-case parameter vector, where the value of the probability density function is maximum among the difference region. It loses precision in modeling the exact level contour proportionally to the decrease in the corresponding probability density function values. Apparently the approximation is very accurate therefore.

- Practical experience shows that the approximation error usually is about 1%–3% absolute yield error.

- The duality principle in minimum norm problems says that the minimum distance of a point to a convex set is equivalent to the maximum distance of the point to all planes that separate the point from the convex set. Figure 45 illustrates the duality principle. In this example, we have assumed without loss of generality that the variances are equal and that the correlations are zero. Then the level contours are spheres.

  On the left side of Figure 45, point (a) is the worst-case point for the given nominal point inside the gray acceptance region. The duality principle in this case applies for the distance of the nominal point to the complement of the acceptance region, which is convex. It says that point (a) provides the linearization among all possible points on the border of the acceptance region from which the nominal point has maximum distance. This is illustrated by point (b), which leads to a linearization with a smaller distance. We can also see that any tangent on the acceptance region's border will lead to an underestimation of the true yield. Therefore the worst-case distance obtained due to point (a) is a greatest lower bound among all tangential approximations of the acceptance region.

On the right side Figure 45, the acceptance region itself is convex now. We can see that point (c) leads to a larger distance between the nominal point and the tangential approximation of the acceptance region's border than the worst-case point (d). Now the duality principle says that the worst-case distance will be the smallest value among all possible distances of the nominal point to tangents of the border of the acceptance region. At the same time, it can be seen that the yield value will be overestimated by such a tangent. Therefore the worst-case distance obtained due to point (d) is a least upper bound among all tangential approximations of the acceptance region.

In summary, the linearization at a worst-case parameter vector provides the best yield approximation among all tangential planes of an acceptance region.

- Figure 38 illustrates that the yield approximation according to (206) and (207) and Table 5 will be very sensitive with regard to the worst-case distance value for yield values around 50%. With increasing yield value this sensitivity decreases, which equally means a smaller error in the yield approximation through worst-case distances.

  For yield values above 99.9% the approximation error becomes negligible.

(184) can be applied to obtain an approximate yield $\bar{Y}_i$ of a performance feature f_i if an upper and a lower performance-specification feature are simultaneously regarded. The resulting value is not equivalent to the yield value that results from the approximate acceptance region as illustrated in Figure 44(c), as long as the performance gradients in the two worst-case parameter vectors are not parallel. Nevertheless, $\bar{Y}_i$ according to (184) is a valuable approximation in the general worst-case analysis as well.

The yield approximation as described in this section assumes that worst-case parameter vectors are on the border of the acceptance region, which usually happens as practical experience shows.

If a performance function is unimodal and the worst-case parameter vector is inside the tolerance region, then the worst-case performance value represents the individual optimum for the considered parameter space. Hence, no performance value beyond the worst-case value is achievable in the considered parameter space and the corresponding yield is 100%.

5.5.2 Realistic Worst-Case Analysis as Special Case

As a linear performance function is a special case of a general nonlinear function, the realistic worst-case analysis is a special case of the general worst-case analysis.

A numerical optimization algorithm can identify if the performance function is linear during the optimization process. It then automatically terminates the optimization process and determines the worst-case performance values and worst-case parameter vectors.

An iterative deterministic solution algorithm for a general worst-case analysis is starting from a sensitivity analysis of the performance features with regard to the parameters. In the first iteration step, a general worst-case analysis then behaves like a realistic worst-case analysis. The realistic worst-case analysis can therefore be interpreted as the first step of a general worst-case analysis.

5.6 Exercise

Given is a single performance function of two parameters:

$$f = x_{s,1} \cdot x_{s,2} \tag{214}$$

f could be for instance the time constant of the RC circuit in Chapter 2. The nominal parameter vector is:

$$\mathbf{x}_{s,0} = [1\ 1]^T \tag{215}$$

The parameters are normally distributed with the covariance matrix:

$$\mathbf{C} = \begin{bmatrix} 0.2^2 & 0 \\ 0 & 0.8^2 \end{bmatrix} \tag{216}$$

Two performance-specification features are given:

$$f \geq f_L \equiv 0.5 \tag{217}$$

$$f \leq f_U \equiv 2.0 \tag{218}$$

- Perform a classical worst-case analysis in the three-sigma tolerance box of parameters based on a linear performance model established at the nominal parameter vector. Calculate the worst-case parameter vectors and corresponding worst-case performance values. Compare the worst-case performance values from the linearized performance model with the “simulated” values at the worst-case parameter vectors. Check if the performance specification is satisfied in the worst-case.

- Do the same as a realistic worst-case analysis for an ellipsoid tolerance region with $\beta_W = 3$.
- Do the same as a general worst-case analysis. Apply the optimality conditions (Appendix C) to calculate a solution. Check if the solution can be inside the ellipsoid tolerance region (in that case $\lambda_{WL/U} = 0$ would hold).

Chapter 6

YIELD ANALYSIS

In this chapter, the yield analysis problem formulated in Section 4.8 will be further developed. The integral in the yield definitions (116) and (117), as well as in (137), (138) and (139) cannot be solved analytically, but have to be solved numerically. Two approaches to solve this task will be described in the following.

The first approach is based on statistical estimation by sampling according to the parameter distribution. It leads to the so-called Monte-Carlo analysis, which is a statistical technique for the numerical computation of integrals. It will be explained that the accuracy of the statistical estimation does not depend on the number of parameters and performance features, but on the size of the sample. An acceptable accuracy requires a large number of simulations.

The second approach is based on the partitioning of the performance specification into the individual performance-specification features and a geometrical approximation of the integration problem. The resulting problem has the same form as the general-worst case analysis described in Section 5.4. This approach is in practice more efficient than a Monte-Carlo technique, but its complexity grows proportionally to the number of parameters.

6.1 Statistical Yield Analysis

According to (137), the yield is defined as the expectation value of the acceptance function (134). This acceptance function takes the value of one (134), if a circuit represented by a vector of parameter values satisfies the performance specification (122). According to (138), (139), the yield partition with respect to a lower or upper performance-specification feature is defined as the expectation value of the acceptance function (135), (136). This acceptance function takes the value of one (135), (136), if the parameter vector satisfies the corresponding performance-specification feature (123), (124).

Based on (B.1) in Section B, an estimator for this expectation value and hence for the yield can be formulated as follows:

$$\begin{aligned}\widehat{Y} &= \widehat{\mathrm{E}}\left\{\delta(\mathbf{x})\right\} = \frac{1}{n_{MC}}\sum_{\mu=1}^{n_{MC}}\delta(\mathbf{x}_s^{(\mu)}) = \frac{n_{ok}}{n_{MC}} = \\ &= \frac{\sharp\{\text{accepted sample elements}\}}{\text{sample size}}\end{aligned} \tag{219}$$

$$\widehat{Y}_{L/U,i} = \widehat{\mathrm{E}}\left\{\delta_{L/U,i}(\mathbf{x}_s)\right\} = \frac{1}{n_{MC}}\sum_{\mu=1}^{n_{MC}}\delta_{L/U,i}(\mathbf{x}_s^{(\mu)}) = \frac{n_{ok,L/U,i}}{n_{MC}} \tag{220}$$

$$\mathbf{x}_s^{(\mu)} \sim \mathcal{D}\left(\mathrm{pdf}(\mathbf{x}_s)\right) \ , \quad i = 1,\ldots,n_{MC}$$

In (219), the yield with regard to the complete performance specification is estimated. In (220), the yield partition with regard to an individual performance-specification feature, which is either a lower or upper bound on a performance feature, is estimated.

The basis of the statistical yield estimation is a sample consisting of n_{MC} sample elements $\mathbf{x}_s^{(\mu)}$, $\mu = 1,\ldots,n_{MC}$, which have been generated according to Section 3.8. As stated in Appendix B, the sample elements are assumed to be independently and identically distributed according to the given distribution. This is not strictly true: the computational random number generator produces a deterministic sequence of pseudo-random numbers for a given initial value.

The generation of a sample can be interpreted as a simulation of the statistically varying production process on a higher design level, where the performance vector $\mathbf{f}$ is determined in dependence of parameters $\mathbf{x}_s$. Each sample element $\mathbf{x}_s^{(\mu)}$ is evaluated by numerical simulation. The obtained performance vector $\mathbf{f}(\mathbf{x}_s^{(\mu)})$ is checked with regard to the performance specification and the corresponding value of the acceptance function in (219), (220) can be determined.

Figure 46 illustrates this process for two statistical parameters. On the left side, level contours of the probability density function of normally distributed parameters are given. On the right side, a performance specification with four performance-specification features is given, which lead to the performance acceptance region shaded in gray. The corresponding parameter acceptance region is also indicated by a gray area on the left side. The parameter acceptance region though is generally not given in an analytical form.

A sample according to the given distribution yields a cloud of parameter vectors that corresponds to the probability density function. After simulation of each parameter vector, a corresponding cloud of performance vectors as on the right side of Figure 46 is available. Each performance vector that satisfies

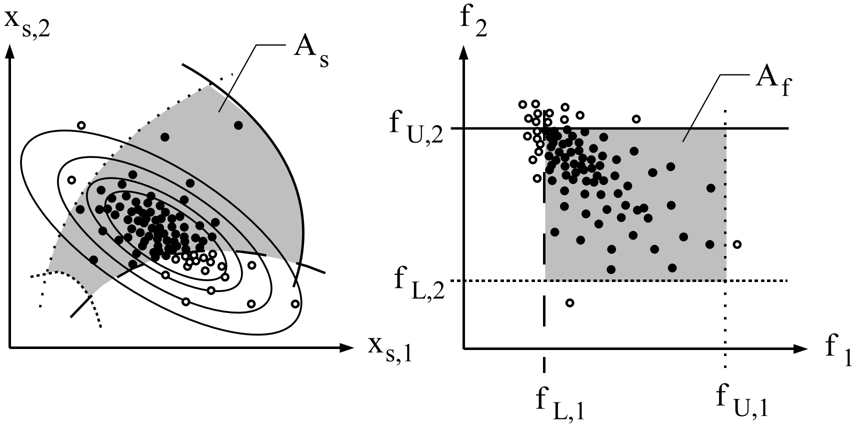


Figure 46. Statistical yield estimation (Monte-Carlo analysis) consists of generating a sample according to the underlying statistical distribution, simulating of each sample element and flagging of elements satisfying the performance specification.

the performance specification is filled black. Likewise each parameter vector can now also be marked according to the corresponding performance vector.

According to (219), the percentage of parameter vectors that are represented in Figure 46 by circles filled black represents the yield estimate.

This statistical estimation is also called Monte-Carlo analysis.

6.1.1 Monte-Carlo Analysis

A Monte-Carlo analysis is a statistical method for the numerical computation of an integral I, which applies the formulation of expectation values (A.1):

$$\begin{aligned} I &= \int \cdots \int_{\mathbf{h}_1(\mathbf{x}) \geq \mathbf{0}} \mathbf{h}(\mathbf{x}) \cdot d\mathbf{x} \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \mathbf{h}_A(\mathbf{x}) \cdot \frac{\mathbf{h}(\mathbf{x})}{\mathrm{pdf}_{MC}(\mathbf{x})} \cdot \mathrm{pdf}_{MC}(\mathbf{x}) \cdot d\mathbf{x} \\ &= \underset{\mathrm{pdf}_{MC}(\mathbf{x})}{\mathrm{E}} \left\{ \mathbf{h}_A(\mathbf{x}) \cdot \frac{\mathbf{h}(\mathbf{x})}{\mathrm{pdf}_{MC}(\mathbf{x})} \right\} \end{aligned} \tag{221}$$

$$\mathbf{h}_A(\mathbf{x}) = \begin{cases} 1, & \mathbf{h}_1(\mathbf{x}) \geq \mathbf{0} \\ 0, & \mathbf{h}_1(\mathbf{x}) < \mathbf{0} \end{cases}$$

In (221), the integrand has been extended by a probability density function pdf_{MC} that is used to scan the integration region. In addition, the integration is done over the complete parameter space, and an indicator function $\mathbf{h}_A$ is introduced in the integrand to include the integration region.

Yield analysis is a special case of Monte-Carlo analysis, where the probability density function is predetermined by the manufacturing tolerances, and where the indicator function is determined by the performance specification and the corresponding acceptance function.

The Monte-Carlo analysis performs a statistical estimation of (221) based on (B.1):

$$\begin{aligned} \widehat{I} &= \underset{\mathrm{pdf}_{MC}(\mathbf{x})}{\widehat{\mathrm{E}}} \left\{ \mathbf{h}_A(\mathbf{x}) \cdot \frac{\mathbf{h}(\mathbf{x})}{\mathrm{pdf}_{MC}(\mathbf{x})} \right\} \\ &= \frac{1}{n_{MC}} \sum_{\mu=1}^{n_{MC}} \mathbf{h}_A(\mathbf{x}^{(\mu)}) \, \frac{\mathbf{h}(\mathbf{x}^{(\mu)})}{\mathrm{pdf}_{MC}(\mathbf{x}^{(\mu)})} \qquad (222) \\ & \mathbf{x}^{(\mu)} \sim \mathcal{D}\left(\mathrm{pdf}_{MC}(\mathbf{x})\right), \quad \mu = 1, \ldots, n_{MC} \end{aligned}$$

The choice of the probability density function is important for the estimation. But even if it is predetermined as in a yield analysis, another distribution may be better for the yield estimation with a Monte Carlo analysis. This is called importance sampling.

6.1.2 Importance Sampling

From Figure 46, we would intuitively expect that a good yield estimation scans more or less the whole parameter acceptance region. This requires a sufficient spread of the sample elements. If the sampling distribution leads to an equal number of sample elements inside and outside of the parameter acceptance region, we would have a yield of 50% and the required even spread of sample elements.

If an estimation, in this case of the yield, is done using a sampling distribution represented by pdf_{MC} that is different from the parameter distribution represented by pdf, this is called importance sampling:

$$\begin{aligned} Y &= \underset{\mathrm{pdf}(\mathbf{x}_s)}{\mathrm{E}} \{\delta(\mathbf{x}_s)\} = \int_{-\infty}^{\infty} \ldots \int_{-\infty}^{\infty} \delta(\mathbf{x}_s) \cdot \mathrm{pdf}(\mathbf{x}_s) \cdot d\mathbf{x}_s \\ &= \int_{-\infty}^{\infty} \ldots \int_{-\infty}^{\infty} \delta(\mathbf{x}_s) \cdot \frac{\mathrm{pdf}(\mathbf{x}_s)}{\mathrm{pdf}_{MC}(\mathbf{x}_s)} \cdot \mathrm{pdf}_{MC}(\mathbf{x}_s) \cdot d\mathbf{x}_s \end{aligned}$$

$$= \underset{\text{pdf}_{MC}(\mathbf{x}_s)}{\mathrm{E}} \left\{ \delta(\mathbf{x}_s) \cdot \frac{\text{pdf}(\mathbf{x}_s)}{\text{pdf}_{MC}(\mathbf{x}_s)} \right\} \tag{223}$$

$$\begin{aligned} \widehat{Y} &= \underset{\text{pdf}_{MC}(\mathbf{x}_s)}{\widehat{\mathrm{E}}} \left\{ \delta(\mathbf{x}_s) \cdot \frac{\text{pdf}(\mathbf{x}_s)}{\text{pdf}_{MC}(\mathbf{x}_s)} \right\} \\ &= \frac{1}{n_{MC}} \sum_{\mu=1}^{n_{MC}} \delta(\mathbf{x}_s^{(\mu)}) \, \frac{\text{pdf}(\mathbf{x}_s^{(\mu)})}{\text{pdf}_{MC}(\mathbf{x}_s^{(\mu)})} \end{aligned} \tag{224}$$

$$\mathbf{x}_s^{(\mu)} \sim \mathcal{D}\left(\text{pdf}_{MC}(\mathbf{x}_s)\right) \ , \quad i = 1, \ldots, n_{MC}$$

The change of the original distribution, for which the expectation value of a function is to be computed, to another sampling distribution leads to a weighting of the function with the ratio of the two distributions.

Importance sampling allows to use arbitrary sampling distributions for the yield estimation in order to improve the quality of the yield estimation.

A measure of the yield estimation quality is the variance of the yield estimation value.

6.1.3 Yield Estimation Accuracy

The variance of the yield estimator can be derived using Appendices A and B:

$$\begin{aligned} \sigma^2_{\widehat{\mathrm{Y}}} = \mathrm{V}\{\widehat{\mathrm{Y}}\} &= \mathrm{V}\{\widehat{\mathrm{E}}\,\{\delta(\mathbf{x}_s)\}\} \\ &\overset{(B.14)}{=} \frac{1}{n_{MC}} \cdot \mathrm{V}\,\{\delta(\mathbf{x}_s)\} \\ &\overset{(A.14)}{=} \frac{1}{n_{MC}} \cdot \big(\mathrm{E}\{\delta^2(\mathbf{x}_s)\} - \mathrm{E}^2\{\delta(\mathbf{x}_s)\}\big) \\ &\overset{\delta^2=\delta}{=} \frac{1}{n_{MC}} \cdot (Y - Y^2) \\ &= \frac{Y \cdot (1 - Y)}{n_{MC}} \end{aligned} \tag{225}$$

(225) is the yield estimator variance, if the yield Y is known. As the yield is usually estimated, $\widehat{Y}$, the formula for an estimator of the yield estimator variance is:

$$\widehat{\sigma}^2_{\widehat{\mathrm{Y}}} = \widehat{\mathrm{V}}\{\widehat{\mathrm{Y}}\} = \widehat{\mathrm{V}}\{\widehat{\mathrm{E}}\,\{\delta(\mathbf{x}_s)\}\}$$

$$\begin{aligned}
&\overset{(B.15)}{=} \quad \frac{1}{n_{MC}} \cdot \widehat{\mathrm{V}}\left\{\delta(\mathbf{x}_s)\right\} \\
&\overset{(B.18)}{=} \quad \frac{1}{n_{MC}} \cdot \frac{n_{MC}}{n_{MC}-1} \cdot \left(\widehat{\mathrm{E}}\{\delta^2(\mathbf{x}_s)\} - \widehat{\mathrm{E}}^{\,2}\{\delta(\mathbf{x}_s)\}\right) \\
&\overset{\delta^2=\delta}{=} \quad \frac{1}{n_{MC}-1} \cdot (\widehat{Y} - \widehat{Y}^2) \\
&= \quad \frac{\widehat{Y} \cdot (1-\widehat{Y})}{n_{MC}-1} \qquad (226)
\end{aligned}$$

(225) and (226) show that the accuracy of a statistical yield estimation by Monte-Carlo analysis primarily depends on the size of the sample, but also on the yield value itself.

Obviously, the variance of the yield estimator $\sigma^2_{\widehat{Y}}$ is quadratical in the yield value. From the first and second-order optimality conditions (Appendix C),

$$\frac{\partial \sigma^2_{\widehat{Y}}}{\partial Y} = \frac{1}{n_{MC}}(1-2Y) \equiv 0 \qquad (227)$$

$$\frac{\partial^2 \sigma^2_{\widehat{Y}}}{\partial Y^2} = -\frac{2}{n_{MC}} < 0 \qquad (228)$$

follows that the variance of the yield estimator has a maximum for a yield of 50%:

$$\arg\max_{Y} \sigma_{\widehat{Y}} = 50\% \qquad (229)$$

The variance of the statistical yield estimation corresponds to the sensitivity of the yield with respect to a shift in the integration bound as illustrated in Figure 20. The hyperbolic-tangent-like shape of the yield function is typical. Figure 20 shows a maximum sensitivity of the yield with regard to a shift in the integration bound if the yield is 50%. For smaller or larger yield values this sensitivity decreases down to zero for a yield of 0% or 100%.

More important is the dependence of the yield estimator variance on the number of sample elements. Table 7 evaluates (225) for an assumed yield of 85% for different sample sizes. A yield of 85% is a typical value in practice and leads to smaller yield estimator variances than a yield of 50% would do.

From Table 7 follows for instance that the three-sigma interval of the estimated yield value is $[\,70\% \ldots 100\%\,]$ for a sample size of 50. For a sample size of 1000, the three-sigma interval of the estimated yield value is $[\,81.7\% \ldots 88.3\%\,]$, which is still quite large.

Let us have a look at the number of sample elements that is required to estimate a yield of 85% with a certain confidence level. For that we will assume

Table 7. Standard deviation of the yield estimator if the yield is 85% for different sample sizes evaluated according to (225).

n_{MC}	10	50	100	500	1000
$\sigma_{\widehat{Y}}$	11.3%	5.0%	3.6%	1.6%	1.1%

that the yield estimate is normally distributed. This is an approximation: as the sample elements are independently and identically distributed and as every sample element has the same probability Y of being in full working order, the number of sample elements that are in full working order originally is binomially distributed. But for an increasing sample size $n_{MC} \rightarrow \infty$, the binomial distribution asymptotically approaches a normal distribution. The binomial distribution can be approximated under easy conditions. For instance, the number of sample elements satisfying the performance specification and the number of sample elements missing the performance specification should each be larger than 4, and the number of total sample elements should be at least 10. Considering these constraints concerning the sample, we can assume that the estimated yield value is normally distributed with the variance (225).

Based on the normal distribution of the yield estimate with variance (225), we can compute the sample size n_{MC} that is required for a yield to be within an interval of $\pm\Delta Y$ around the estimated value $\widehat{Y}$ with a confidence level of $\gamma[\%]$ in the following way.

The confidence level γ denotes the probability that the estimated value is within the given interval. This probability corresponds to an interval that is described by a multiple k_γ of the underlying yield variance $\widehat{\sigma}_{\widehat{Y}}$. Using (16)–(19), k_γ can be determined as the value that satisfies:

$$\gamma = \mathrm{cdf}(\widehat{Y} + k_\gamma \cdot \widehat{\sigma}_{\widehat{Y}}) - \mathrm{cdf}(\widehat{Y} - k_\gamma \cdot \widehat{\sigma}_{\widehat{Y}}) \ \rightarrow \ k_\gamma \tag{230}$$

k_γ is determined without knowing the yield estimate and yield estimate variance through the corresponding normalized univariate normal distribution.

For instance, a confidence level of $\gamma = 90\%$ denotes a probability of 90% for the yield estimate to be within an interval of $\pm 1.645\widehat{\sigma}_{\widehat{Y}}$ around the estimated value $\widehat{Y}$. A confidence level of $\gamma = 95\%$ denotes a probability of 95% for the yield estimate to be within an interval of $\pm 1.960\widehat{\sigma}_{\widehat{Y}}$ around the estimated value $\widehat{Y}$. $\gamma = 99\%$ corresponds to $\pm 2.576\widehat{\sigma}_{\widehat{Y}}$, and $\gamma = 99.9\%$ corresponds to $\pm 3.291\widehat{\sigma}_{\widehat{Y}}$.

Table 8. Required sample size for an estimation of a yield of 85% for different confidence intervals and confidence levels according to (232).

Confidence level	90%	95%	99%	99.9%
Confidence $k_\gamma \widehat{\sigma}_{\widehat{Y}} \rightarrow$ interval $\downarrow Y \pm \Delta Y$	$\pm 1.645 \sigma_{\widehat{Y}}$	$\pm 1.960 \sigma_{\widehat{Y}}$	$\pm 2.576 \sigma_{\widehat{Y}}$	$\pm 3.291 \sigma_{\widehat{Y}}$
$85\% \pm 10\%$	35	49	85	139
$85\% \pm \ \ 5\%$	139	196	339	553
$85\% \pm \ \ 1\%$	3,451	4,899	8,461	13,810

The confidence interval is now given by two equal forms:

$$\Delta Y = k_\gamma \cdot \widehat{\sigma}_{\widehat{Y}} \tag{231}$$

Inserting the obtained value k_γ and (231) in (225), the required sample size n_{MC} for the given confidence level $\gamma \mapsto k_\gamma$ and interval $\widehat{Y} \pm \Delta Y$ can be computed:

$$n_{MC} \approx \frac{Y \cdot (1 - Y) \cdot k_\gamma^2}{\Delta Y^2} \tag{232}$$

For a yield of 85%, Table 8 evaluates (232) for three different confidence intervals, $75\% \ldots 95\%$, $80\% \ldots 90\%$, and $84\% \ldots 86\%$, and for four different confidence levels, 90%, 95%, 99%, and 99.9%.

Table 8 shows for instance that a Monte Carlo analysis with 8,461 sample elements is required if we want have a 99% probability that the yield is within $\pm 1\%$ around its estimated value of 85%. It can be seen that thousands of sample elements are required for a sufficient accuracy of the statistical yield estimation. According to (225), to increase the accuracy by a factor of F, the sample size has to be increased by a factor of F^2. This is illustrated in Table 8, where the increase in accuracy by a factor of 10 from $\Delta Y = 10\%$ to $\Delta Y = 1\%$ corresponds to an increase in the sample size by a factor of 100.

Each sample element has to be evaluated by simulation. Simulation is computationally very expensive and exceeds by far the computational cost of the remaining operations. The computational cost of a Monte Carlo analysis is therefore mainly determined by the number of simulations, i.e. the sample size.

Overall we have for a Monte-Carlo analysis:

$$\begin{aligned} \text{Accuracy} &\sim \sqrt{n_{MC}} \\ \text{Complexity} &\sim n_{MC} \end{aligned}$$

Interestingly, the accuracy does not depend on any other quantity like for instance the nonlinearity of the performance, and the complexity does not depend on any other quantity like for instance the number of parameters.

6.2 Tolerance Classes

An alternative to a statistical yield analysis is a geometric approach. A certain type of tolerance region of the statistical parameters is assumed and a maximum size of this tolerance region type within the given parameter acceptance region is computed. The yield is derived from the size of the tolerance region. The specific relation between tolerance region T_s and yield Y is denoted as tolerance class. In the following, some types of tolerance classes will be treated.

We will assume that the statistical parameters are normally distributed or have been transformed into normally distributed parameters.

6.2.1 Tolerance Interval

If there is only one statistical parameter x_s, which is normally distributed with mean value $x_{s,0}$ and variance σ^2, $x_s \sim \mathcal{N}(x_{s,0}, \sigma^2)$, the definition of a tolerance class is straightforward:

$$T_{s,I} = \{x_s \mid a \le x_s \le b\} \tag{233}$$

$$\begin{aligned} \leftrightarrow Y_I &= \int_a^b \mathrm{pdf}_N(x_s) \cdot dx_s \\ &= \mathrm{cdf}_N(b) - \mathrm{cdf}_N(a) \qquad (234) \\ &= \int_{\beta_a}^{\beta_b} \frac{1}{\sqrt{2\pi}} \, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt \qquad (235) \end{aligned}$$

$$\beta_a = \frac{a - x_{s,0}}{\sigma}\ , \ \ \beta_b = \frac{b - x_{s,0}}{\sigma}\ , \ \ t \sim \mathcal{N}(0,1)$$

pdf_N denotes the probability density function of the univariate normal distribution as defined in (16). cdf_N denotes the corresponding cumulative distribution function, which is defined in (17)–(19).

Based on (41), the probability density function is transformed into a probability density function with a mean value of zero and a variance of one (235). In this way, the integration bounds a and b have been transformed into their

Table 9. Tolerance intervals $T_{s,I}$ and corresponding yield values Y_I according to (235).

$T_{s,I} = [\beta_a\sigma, \beta_b\sigma]$	Y_I
$[\,-1\sigma\,,\,1\sigma\,]$	68.3%
$[\,-2\sigma\,,\,2\sigma\,]$	95.5%
$[\,-3\sigma\,,\,3\sigma\,]$	99.7%

$T_{s,I} = [\beta_a\sigma, \beta_b\sigma]$	Y_I
$[\,-\infty\,,\,-1\sigma\,]$	15.9%
$[\,-\infty\,,\,0\,]$	50.0%
$[\,-\infty\,,\,1\sigma\,]$	84.1%
$[\,-\infty\,,\,2\sigma\,]$	97.7%
$[\,-\infty\,,\,3\sigma\,]$	99.9%

distances to the nominal value as multiples of the standard deviation. Note that the resulting integration bounds β_a, β_b represent the univariate case of a worst-case distance defined in (24), (161) or (185).

Similar to Table 5 and Figure 20, values of tolerance intervals $T_{s,I}$ and corresponding yield values Y_I can be taken from statistical tables. Table 9 shows some examples. We can see the familiar probabilities of a univariate normal distribution. In the left column, symmetric intervals around the mean value are presented. In the right column, parameter values bounded from the right are shown. We can see that worst-case distances may be supplemented with a negative sign in order to indicate that the mean value is outside the tolerance interval.

6.2.2 Tolerance Box

If more than one parameter is there, the concept of intervals can be extended to the multivariate case of a normally distributed parameter vector $\mathbf{x}_s \sim \mathcal{N}(\mathbf{x}_{s,0}, \mathbf{C})$. This results in tolerance boxes as defined in (8). Tolerance boxes are also the given tolerance regions in the classical worst-case analysis described in Section 5.1. Analogously to (234) and (235), the tolerance box $T_{s,B}$ and corresponding yield Y_B are defined:

$$T_{s,B} \;=\; \{\mathbf{x}_s \,|\, \mathbf{a} \leq \mathbf{x}_s \leq \mathbf{b}\} \tag{236}$$

$$\leftrightarrow Y_B \;=\; \int_{a_1}^{b_1} \ldots \int_{a_{n_{xs}}}^{b_{n_{xs}}} \mathrm{pdf}_N(\mathbf{x}_s)\cdot d\mathbf{x}_s$$

$$= \int_{\beta_{a_1}}^{\beta_{b_1}} \ldots \int_{\beta_{a_{n_{xs}}}}^{\beta_{b_{n_{xs}}}} \frac{1}{\sqrt{2\pi}^{\,n_{xs}}} \exp\left(-0.5\cdot \mathbf{t}^T\cdot \mathbf{R}^{-1}\cdot \mathbf{t}\right)\cdot d\mathbf{t} \tag{237}$$

$$\mathbf{t} = \mathbf{\Sigma}^{-1}\cdot(\mathbf{x}_s - \mathbf{x}_{s,0})\;,\;\; \mathbf{t} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}) \tag{238}$$

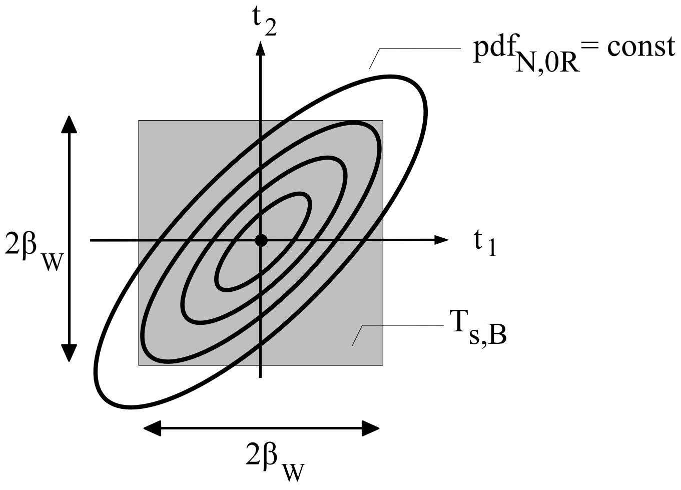


Figure 47. Tolerance box $T_{s,B}$ and normal probability density function $\text{pdf}_{N,0R}$ with zero mean and unity variance.

$$\beta_{a_k} = \frac{a_k - x_{s,0,k}}{\sigma_k} \ , \ \beta_{b_k} = \frac{b_k - x_{s,0,k}}{\sigma_k} \tag{239}$$

pdf_N denotes the probability density function of the multivariate normal distribution as defined in (23)–(29). Using the variable transformation (238) and the decomposition (26) of the covariance matrix, we obtain parameters $\mathbf{t}$ that are normally distributed each with a mean value of 0 and a variance of 1, and that are mutually correlated with the correlation matrix $\mathbf{R}$. The probability density function of the resulting normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{R})$ is denoted as $\text{pdf}_{N,0R}$.

The variable transformation (238) also transforms the integration bounds a_k and b_k into their distances to their nominal values as multiples of their standard deviations, β_{a_k} and β_{b_k}.

(237) represents an integral over the probability density function $\text{pdf}_{N,0R}$ in a box $T_{s,B}$. Figure 47 illustrates level contours of $\text{pdf}_{N,0R}$ and $T_{s,B}$ for two parameters. In this example, $\beta_{a_k} = -\beta_{b_k} = -\beta_W$ has been chosen. The resulting box is symmetrical around the origin.

(237) has to be solved numerically. Only if the parameters are uncorrelated, (237) can be evaluated by using the cumulative distribution function values of the univariate normal distribution:

$$\mathbf{R} = \mathbf{I} \ : \qquad Y_B = \prod_{k=1}^{n_{xs}} \int_{\beta_{a_k}}^{\beta_{b_k}} \frac{1}{\sqrt{2\pi}} \, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt \tag{240}$$

Table 10. Yield values Y_B for a tolerance box $T_{s,B}$ with $\forall_k\ \beta_{a_k} = -\beta_{b_k} = -\beta_W = -3$ in dependence of the number of parameters n_{xs} and of the correlation. $\forall_{k \neq l}\ \varrho_{k,l} = 0.0$ according to (241), $\forall_{k \neq l}\ \varrho_{k,l} = 0.8$ according to (237).

n_{xs}	$Y_B\vert_{\varrho_{k,l}=0.0}$	$Y_B\vert_{\varrho_{k,l}=0.8}$
3	99.1%	99.3%
4	98.8%	99.2%
5	98.5%	99.1%
6	98.2%	99%
7	97.9%	98.9%
8	97.6%	98.85%
9	97.3%	98.8%
10	97.0%	98.7%

$$\left.\begin{array}{r} \mathbf{R} = \mathbf{I} \\ \forall_{k \neq l}\ \beta_{a_k} = \beta_{a_l} = \beta_a \\ \forall_{k \neq l}\ \beta_{b_k} = \beta_{b_l} = \beta_b \end{array}\right\} : \quad Y_B = \left(\int_{\beta_a}^{\beta_b} \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt \right)^{n_{xs}} \qquad (241)$$

Table 10 shows the yield value Y_B corresponding to a tolerance box $T_{s,B}$ with $\forall_k\ \beta_{a_k} = -\beta_{b_k} = -\beta_W = -3$. Y_B is given for different numbers of parameters n_{xs} and two different correlation values. For $\varrho = 0.0$, (241) is applied to compute Y_B, for $\varrho_{k,l} = 0.8$, (237) is applied. The given correlation value holds for all parameter pairs.

It can be seen that the yield Y_B that corresponds to a tolerance box depends on the dimension of the parameter space and on the correlations between parameters. This dependence is more pronounced for smaller tolerance boxes than the one with $\beta_W = 3$ in Table 10.

From Figure 22 follows that a tolerance box represents the worst-case if correlations are unknown. Then, a tolerance box and corresponding yield value according to (237) seems to be appropriate. Usually, at least a qualitative knowledge about parameters being rather uncorrelated or strongly correlated is available. In that case, we are interested to use a tolerance class that does not depend on the number of parameters and does not depend on the correlation values. As the parameters or transformed parameters can be assumed to be normally distributed, it is promising to apply the tolerance region that results from the level contours of the probability density function, i.e. ellipsoids.

6.2.3 Tolerance Ellipsoid

The ellipsoid tolerance class reproduces the equidensity contours of normally distributed parameters $\mathbf{x}_s \sim \mathcal{N}(\mathbf{x}_{s,0}, \mathbf{C})$, which are ellipsoids determined by

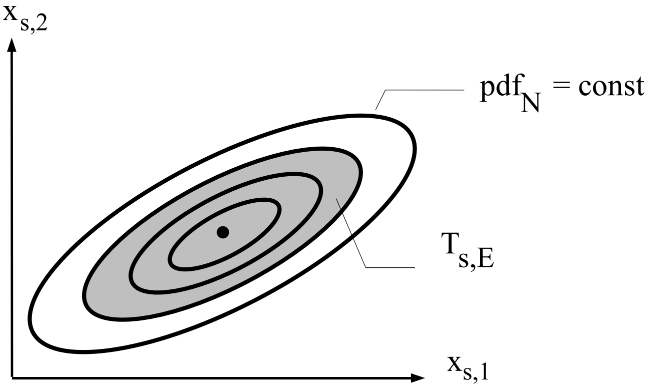


Figure 48. Ellipsoid tolerance region $T_{s,E}$ and equidensity contours of normal probability density function pdf_N.

the quadratic form (24). Defining a tolerance class according to such an ellipsoid, the ellipsoid tolerance region $T_{s,E}$ is:

$$T_{s,E} = \{\mathbf{x}_s \mid (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \leq \beta_W^2\} \qquad (242)$$

Figure 48 illustrates the ellipsoid tolerance region $T_{s,E}$ for two parameters. Ellipsoid tolerance regions are also the given tolerance regions in the realistic worst-case analysis described in Section 5.2 and the general worst-case analysis describe in Section 5.4.

In order to get to the corresponding yield Y_E, we apply the variable transformation (58),

$$\mathbf{t} = \mathbf{A}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}), \qquad (243)$$

and the ellipsoid tolerance region becomes:

$$T_{s,E} = \left\{ \mathbf{t} \mid \mathbf{t}^T \cdot \mathbf{t} \leq \beta_W^2 \right\} = \left\{ \mathbf{t} \;\middle|\; \sum_{k=1}^{n_{xs}} t_k^2 \leq \beta_W^2 \right\} \qquad (244)$$

The transformed variable $\mathbf{t}$ is normally distributed with mean values of zero, variances of one, and correlations of zero:

$$\mathbf{t} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad \Leftrightarrow \quad t_k \sim \mathcal{N}(0,1) \ , \quad k = 1, \ldots, n_{xs} \qquad (245)$$

As the random variables t_k are independently and identically normally distributed, $\beta^2 = \mathbf{t}^T \cdot \mathbf{t} = \sum_{k=1}^{n_{xs}} t_k^2$ is χ^2(chi-square)-distributed with n_{xs} degrees

Table 11. Yield values Y_E for an ellipsoid tolerance region $T_{s,E}$ with $\beta_W = 3$ in dependence of the number of parameters n_{xs} according to (247) and (248).

n_{xs}	Y_E
2	98.9%
3	97.1%
4	93.9%
5	89.1%
6	82.6%
7	74.7%
8	65.8%
9	56.3%
10	46.8%
$\vdots$	$\vdots$
15	12.3%

of freedom:

$$\beta^2 = \sum_{k=1}^{n_{xs}} t_k^2 \quad \sim \quad \chi^2_{n_{xs}} \tag{246}$$

The χ^2(chi-square)-distribution has the following probability density function and is tabulated in statistical handbooks and subroutines.

$$\text{pdf}_{\chi^2_{n_{xs}}}(\beta^2) = \frac{(\beta^2)^{\frac{n_{xs}}{2}-1} \cdot \exp(-\frac{\beta^2}{2})}{2^{\frac{n_{xs}}{2}} \cdot \Gamma(\frac{n_{xs}}{2})} \tag{247}$$

The yield value Y_E corresponding to $T_{s,E}$ then is defined by:

$$T_{s,E} \quad \leftrightarrow \quad Y_E = \int_0^{\beta_W^2} \text{pdf}_{\chi^2_{n_{xs}}}(\beta^2) \cdot d\beta^2 \tag{248}$$

From (247) and (248) we can see that the yield Y_E that corresponds to an ellipsoid tolerance region is independent of the correlation values of the normal distribution of the parameters, contrary to a box tolerance region. But (247) and (248) also show that Y_E depends on the dimension of the parameter space, n_{xs}.

Table 11 shows the yield value Y_E corresponding to a tolerance box $T_{s,E}$ with $\beta_W = 3$ for different number of parameters. Obviously the yield Y_E within a tolerance ellipsoid determined by β_W strongly decreases with an increasing number of parameters. This property is advantageous for the sampling properties of the normal distribution, as it adapts the spreading of sample elements to the dimension of the scanned parameter space. But this property is disadvantageous for a tolerance class, where it is unwanted if a tolerance region refers to different yield values depending on the number of parameters.

6.2.4 Single-Plane-Bounded Tolerance Region

The single-plane-bounded tolerance class cuts the parameter space in two halves by a given single plane.

The motivation for this type of tolerance class is the consideration of a single performance-specification feature. The previous tolerance class types are restricted to reproducing the parameter distribution. Contrary to that, the single-plane-bounded tolerance class considers both the parameter distribution and the performance specification.

Note that the following considerations are similar to those done previously for a worst-case analysis. Yet, there are differences and advances in the following description.

In Sections 5.2 and 5.4, we started from a tolerance region of statistical parameters and computed worst-case performance-feature values. We illustrated that the worst-case distance as an input determining the size of the tolerance region relates to the yield partition of an individual performance feature (Sections 5.3 and 5.5). In the following, we will reverse the direction and start from a performance-specification-feature value and relate it to a yield partition value and corresponding size of tolerance region. While in Sections 5.2 and 5.4, usually the same tolerance region and corresponding worst-case distance is given for all performance features, now different worst-case distances and yield partitions will result for the individual performance-specification features. In addition, as we start from a performance-specification feature, we will face four cases that emanate from having a lower or upper performance-feature bound, which may be violated or satisfied at the nominal parameter vector. Last not least, the following description will be based on geometric considerations, while Sections 5.2 and 5.4 were based on optimality conditions.

In the following, the case of a tolerance region that results from an upper bound f_U on a single performance feature f is described. An index i denoting the performance feature is left out for simplicity. The single-plane-bounded tolerance region $T_{s,SP,U}$ and corresponding yield partition $Y_{SP,U}$ are defined by:

$$T_{s,SP,U} = \{\mathbf{x}_s \mid \mathbf{g}^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,WU}) \leq 0\} \tag{249}$$

$$\leftrightarrow Y_{SP,U} = \int \cdots \int_{T_{s,SP,U}} \mathrm{pdf}_N(\mathbf{x}_s) \cdot d\mathbf{x}_s$$

$$= \int \cdots \int_{\mathbf{g}^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,WU}) \leq 0} \mathrm{pdf}_N(\mathbf{x}_s) \cdot d\mathbf{x}_s \tag{250}$$

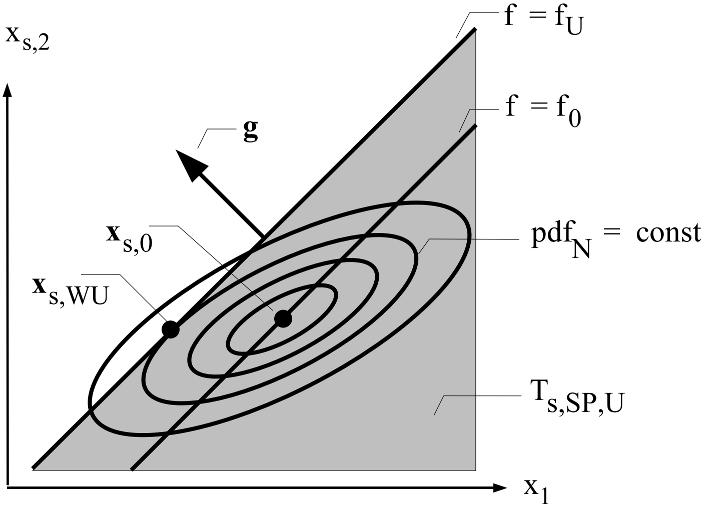


Figure 49. Single-plane-bounded tolerance region $T_{s,SP,U}$ and equidensity contours of normal probability density function pdf_N.

Figure 49 illustrates a single-plane-bounded tolerance region for two parameters. Assuming a linear performance model,

$$f = f_U + \mathbf{g}^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,WU}) \tag{251}$$

the normally distributed parameters $\mathbf{x}_s \sim \mathcal{N}(\mathbf{x}_{s,0}, \mathbf{C})$ are transformed into a normally distributed performance feature:

$$f \sim \mathcal{N}(f_U + \mathbf{g}^T \cdot (\mathbf{x}_{s,0} - \mathbf{x}_{s,WU}), \sigma_f^2) \tag{252}$$

$$\sigma_f^2 = \mathbf{g}^T \cdot \mathbf{C} \cdot \mathbf{g} \tag{253}$$

From (251) follows that

$$f \le f_U \quad \Leftrightarrow \quad \mathbf{g}^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,WU}) \le 0 \tag{254}$$

In Figure 49, $\mathbf{x}_{s,WU}$ being an upper bound means that the gradient $\mathbf{g}$ points from the border of $T_{s,SP,U}$ away from the tolerance region $T_{s,SP,U}$.

We can formulate an equivalent form of the single-plane-bounded tolerance region $T_{s,SP,U}$ and corresponding yield partition $Y_{SP,U}$ to (249) and (250) in the space of the single performance feature f:

$$T_{s,SP,U} = \{\mathbf{x}_s \,|\, f(\mathbf{x}_s) \le f_U\} \tag{255}$$

$$\leftrightarrow Y_{SP,U} \;=\; \int_{-\infty}^{f_U} \mathrm{pdf}_f(f) \cdot df \tag{256}$$

pdf_f is a normal probability density function with mean value and covariance as given in (252) and (253).

Let us express the difference between the given bound f_U and the performance value at $\mathbf{x}_{s,0}$, $f_0 = f(\mathbf{x}_{s,0})$, as a multiple of the performance variance σ_f^2 (253):

$$f_U - f_0 = \mathbf{g}^T \cdot (\mathbf{x}_{s,WU} - \mathbf{x}_{s,0}) \equiv \begin{cases} +\,\beta_{WU} \cdot \sigma_f\,, & f_0 \leq f_U \\ -\,\beta_{WU} \cdot \sigma_f\,, & f_0 \geq f_U \end{cases} \tag{257}$$

(257) considers that the nominal parameter vector $\mathbf{x}_{s,0}$ can be inside or outside of the tolerance region $T_{s,SP,U}$ according to (249). If $\mathbf{x}_{s,0}$ is outside of $T_{s,SP,U}$, then $f_0 > f_U$ and the difference is negative.

(256) can be reformulated for a standardized normal distribution with zero mean and unity variance using (257):

$$Y_{SP,U} \;=\; \begin{cases} \displaystyle\int_{-\infty}^{\beta_{WU}} \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt\,, & f_0 \leq f_U \\ \displaystyle\int_{-\infty}^{-\,\beta_{WU}} \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt\,, & f_0 \geq f_U \end{cases} \tag{258}$$

The equivalence of (249) and (255) means that a single-plane-bounded tolerance region $T_{s,SP,U}$ corresponds to the acceptance region of a single performance-specification feature, i.e. a single bound on a performance feature.

The equivalence of (250) and (258) means that the corresponding yield partition $Y_{SP,U}$ can be evaluated based on the standardized normal distribution with zero mean and unity variance.

The same considerations can be done for a lower bound on a performance feature, starting from the corresponding formulation of the tolerance region, $T_{s,SP,L} = \{\mathbf{x}_s \mid \mathbf{g}^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,WL}) \geq 0\}$. The resulting yield partition is:

$$Y_{SP,L} \;=\; \begin{cases} \displaystyle\int_{-\infty}^{\beta_{WL}} \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt\,, & f_0 \geq f_L \\ \displaystyle\int_{-\infty}^{-\,\beta_{WL}} \frac{1}{\sqrt{2\pi}}\, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt\,, & f_0 \leq f_L \end{cases} \tag{259}$$

From (258) and (259) follows that the worst-case distance gets a positive sign, if the performance-specification feature is satisfied at the nominal parameter vector, and gets a negative sign, if the performance-specification feature is violated at the nominal parameter vector. Figure 50 illustrates the four cases

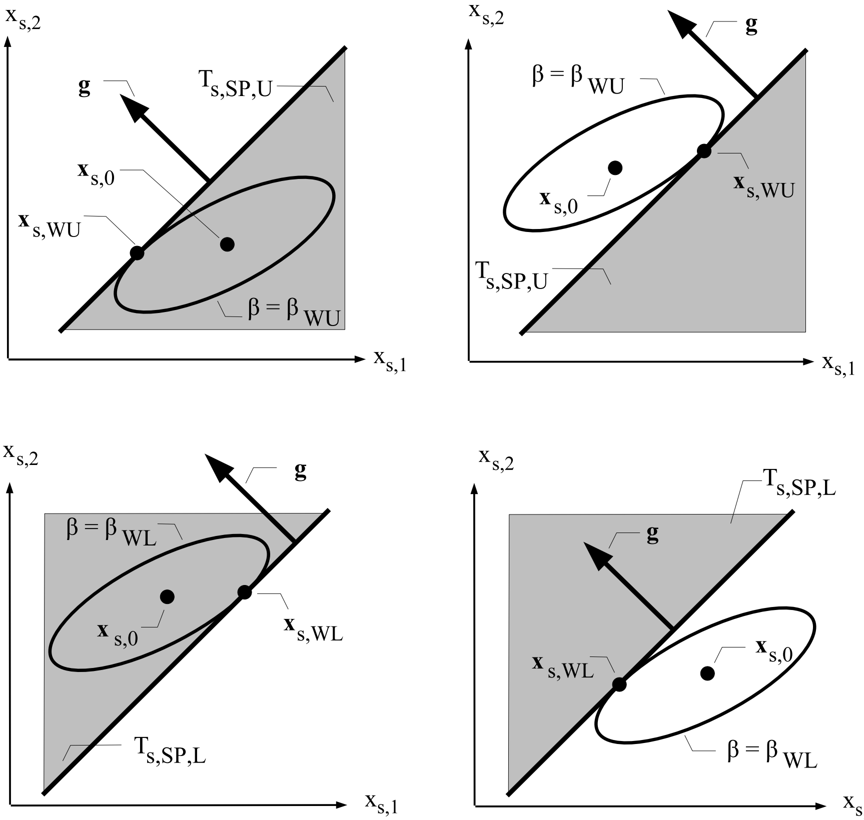


Figure 50. Single-plane-bounded tolerance regions $T_{s,SP,L}$, $T_{s,SP,U}$ for a single performance feature with either an upper bound (first row) or a lower bound (second row), which is either satisfied (first column) or violated at the nominal parameter vector (second column).

that result from a lower or upper performance-feature bound violated or satisfied by the nominal design.

Table 12 shows selected yield partition values Y_{SP} for some selected single-plane-bounded tolerance regions T_{SP} regardless if it originates from a lower or upper performance-feature bound.

Obviously the single-plane-bounded tolerance class provides yield partition values that can easily be evaluated using statistical tables and functions and that additionally are independent of the number of parameters and of the variances and correlations.

Table 12. Single-plane-bounded tolerance region $T_{s,SP}$ and corresponding yield partition values Y_{SP} according to (258) or (259).

$T_{s,SP} = [\,-\infty\,,\,\beta_W \sigma_f\,]$	Y_{SP}
$[\,-\infty\,,\,-1\sigma_f\,]$	15.9%
$[\,-\infty\,,\,0\,]$	50.0%
$[\,-\infty\,,\,1\sigma_f\,]$	84.1%
$[\,-\infty\,,\,2\sigma_f\,]$	97.7%
$[\,-\infty\,,\,3\sigma_f\,]$	99.9%

The single-plane-bounded tolerance class is suitable if the performance specification is partitioned into the individual performance-specification features, which leads to a partitioning of the parameter acceptance region as illustrated in Figure 32.

It can also be shown that the single-plane-bounded tolerance region defined by β_W determines a corresponding ellipsoid tolerance region. This is done in the following.

Figure 49 indicates that there is an equidensity contour of the parameters' probability density function that touches the plane (249) at a certain point, for instance at $\mathbf{x}_W$. This means that the orthogonal on the quadratic form (24) at $\mathbf{x}_W$ is parallel to $\mathbf{g}$:

$$\mathbf{C}^{-1} \cdot (\mathbf{x}_{s,WU} - \mathbf{x}_{s,0}) = \begin{cases} +\,\lambda \cdot \mathbf{g}\,, & f_0 \le f_U \\ -\,\lambda \cdot \mathbf{g}\,, & f_0 \ge f_U \end{cases} \tag{260}$$

Inserting $\mathbf{x}_{s,WU} - \mathbf{x}_{s,0}$ from (260) in the equivalence part of (257) results in:

$$\lambda = \frac{\beta_{WU}}{\sigma_f} \tag{261}$$

Solving (260) for $\mathbf{x}_{s,WU} - \mathbf{x}_{s,0}$ and inserting λ from (261), and inserting the resulting expression for $\mathbf{x}_{s,WU} - \mathbf{x}_{s,0}$ into (24) results in

$$(\mathbf{x}_{s,WU} - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_{s,WU} - \mathbf{x}_{s,0}) = \beta_{WU}^2 \tag{262}$$

(262) shows that the β_W-multiple of the difference between nominal performance and performance bound corresponds to the tolerance ellipsoid touching the given plane.

From Sections 5.2 and 5.5, it is known that the realistic and general worst-case analysis, if they are based on the single-plane-bounded tolerance class,

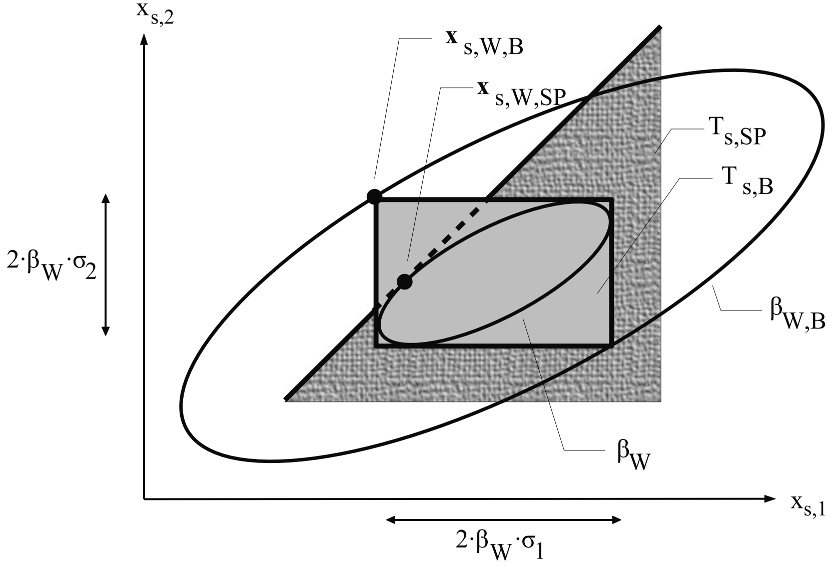


Figure 51. Single-plane-bounded tolerance region $T_{s,SP}$ with corresponding worst-case parameter vector $\mathbf{x}_{W,SP}$ and worst-case distance β_W for two parameters. β_W denotes a $\pm\beta_W$ times the covariances tolerance ellipsoid. Tolerance box $T_{s,B}$ determined by $\pm\beta_W$ times the covariances with corresponding worst-case parameter vector $\mathbf{x}_{W,B}$ and worst-case distance $\beta_{W,B}$.

lead to a high accuracy of the approximated yield partition value of a single performance-specification feature. Among the mentioned tolerance classes, the yield value of a single performance-specification feature $f_i \geq f_{L,i}$ or $f_i \leq f_{U,i}$ is therefore geometrically optimally approximated based on a single-plane-bounded tolerance region, i.e. according to Table 12.

6.2.5 Corner Worst Case vs. Realistic Worst Case

Corner parameter vectors obtained from a classical worst-case analysis on the other hand result in exaggerated robustness estimations. Figure 51 illustrates the resulting worst-case parameter vectors and yield approximations of a classical worst-case analysis and a realistic worst-case analysis.

A classical worst-case analysis is based on a tolerance box of parameters, the corresponding quantities in Figure 51 therefore have the index B. A realistic worst-case analysis is based on a single-plane-bounded tolerance region, the corresponding quantities in Figure 51 therefore have the index SP.

Both types of worst-case analysis are based on a tolerance region referring to the same multiple of the respective parameter variances. The realistic

Table 13. Exaggerated robustness $\beta_{W,B}$ represented by a corner worst-case parameter vector of a classical worst-case analysis, for different correlation $\forall_{k \neq l}\, \varrho_{k,l} = \varrho$ among the parameters, and for different numbers of parameters n_{xs}.

$\varrho \rightarrow$ $\downarrow n_{xs}$	0	0.3	0.6	0.9
2	$4.2\sigma_f$	$5.1\sigma_f$	$6.7\sigma_f$	$13.4\sigma_f$
3	$5.2\sigma_f$	$6.1\sigma_f$	$7.8\sigma_f$	$15.5\sigma_f$
4	$6.0\sigma_f$	$7.2\sigma_f$	$9.5\sigma_f$	$19.0\sigma_f$
		$\beta_W = 3\sigma_f$		

worst-case analysis starts from an ellipsoid determined by β_W. This ellipsoid is within a range of $\pm\beta_W \cdot \sigma_k$ around the nominal parameter values $x_{0,k}$. The combination of these intervals of the individual parameters determines the tolerance box $T_{s,B}$ of a classical worst-case analysis. Note that $T_{s,B}$ holds for any correlation according to (37) and as illustrated in Figure 22(d).

Due to these initial tolerance regions, the resulting worst-case parameter vector $\mathbf{x}_{W,B}$ of a classical worst-case analysis will always refer to larger ellipsoid and worst-case distance than the resulting worst-case parameter vector $\mathbf{x}_{W,SP}$ of a realistic worst-case analysis:

$$\beta_{W,B} \equiv \beta(\mathbf{x}_{s,W,B}) \quad \geq \quad \beta_W \equiv \beta(\mathbf{x}_{s,W,SP}) \tag{263}$$

The robustness that is represented by a corner worst case therefore is always more exaggerated than that represented by a realistic worst case. The degree of exaggeration depends on the number of parameters, on the correlation and on the correlation in relation to the performance sensitivity.

Table 13 illustrates how exaggerated the robustness represented by the corner worst case can be. The underlying classical worst-case analysis has a tolerance box $\pm 3\sigma_k$ around the nominal parameter vector, which refers to a worst-case distance of $\beta_W = 3$ and a yield of $Y_W = 99.9\%$ in a realistic worst-case analysis.

The entry in the last row and last column for instance says that for 4 parameters and a correlation of 0.9 among the parameters, the corner worst case obtained from a $\pm 3\sigma_k$ may correspond to a safety margin of $\beta_{W,B} = 19.0\sigma$, which is excessive.

6.3 Geometric Yield Analysis

Unlike a statistical yield analysis, which is based on statistical sampling of the parameter space, a geometric yield analysis is based on an approximation of the parameter acceptance region.

A geometric approximation of the parameter acceptance region is more easy if a partitioning of the performance specification into the individual performance-specification features is done. This partitioning of the performance specification according to lower and upper bounds of individual performance features (123)–(125) has been illustrated in Figure 32.

As before, we assume that the statistical parameters are normally distributed or have been transformed into normally distributed parameters.

6.3.1 Problem Formulation

We can distinguish between four cases of a geometric yield analysis, which arise because either a lower or an upper bound on the performance feature is given, which is either satisfied or violated at the nominal parameter vector.

Figure 52 picks the lower bound $f_{L,1}$ of performance f_1 in Figure 32 and illustrates the geometric analysis for the case that a lower bound f_L on a performance feature f is given, and that the nominal design $\mathbf{x}_{s,0}$ is inside the parameter acceptance region partition. To simplify the description, the index i denoting the ith performance feature will be left out.

Figure 52 shows the border of the parameter acceptance region partition $A_{s,L}$, which is generally nonlinear. This nonlinear border cannot be formulated analytically due to the nature of performance functions in analog design. An approximation of the border shall be formulated that determines its most important point and the corresponding performance gradient.

As a larger ellipsoid corresponds to a smaller probability density value in Figure 52, it becomes visible that the statistical parameter vectors outside of the acceptance region partition $A_{s,L}$ or just on the border of $A_{s,L}$ differ from each other concerning their probability density value. Among all points outside or on the border of the acceptance region, there exists a point that has a maximum probability density value. The infinitesimal region around this parameter vector has therefore the highest probability of occurrence among all parameter vectors on the border. We call this distinguished parameter vector "worst-case parameter vector" and denote it as $\mathbf{x}_{s,WL}$.

If the nominal design would violate the lower bound on the performance feature, then the center $\mathbf{x}_{s,0}$ of the ellipsoidal equidensity contours in Figure 52 would be outside of the parameter acceptance region partition, that means on the non-gray side. In this case as well, there is a distinguished parameter vector with highest probability of occurrence, but this time among all parameter vectors inside or on the border of the parameter acceptance region partition.

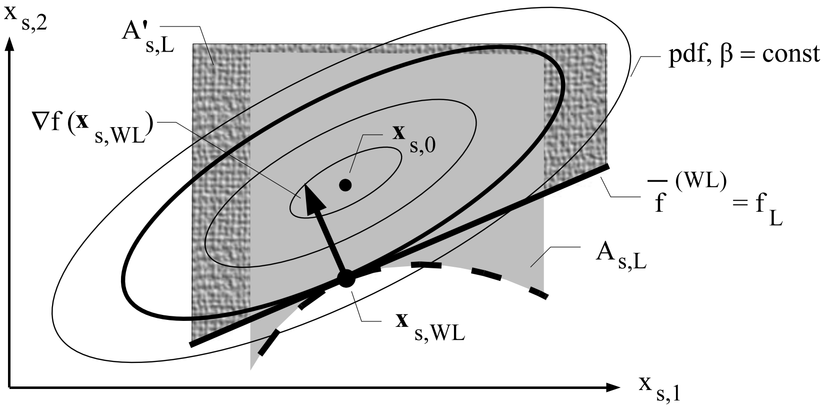


Figure 52. Geometric yield analysis for a lower bound on a performance feature, $f > f_L$, which is satisfied at the nominal parameter vector. Worst-case parameter vector $\mathbf{x}_{s,WL}$ has the smallest distance from the nominal statistical parameter vector $\mathbf{x}_{s,0}$ measured according to the equidensity contours among all parameter vectors outside of or on the border of the acceptance region partition $A_{s,L}$. The tangential plane to the tolerance ellipsoid through $\mathbf{x}_{s,WL}$ as well as to the border of $A_{s,L}$ at $\mathbf{x}_{s,WL}$ determines a single-plane-bounded tolerance region $A'_{s,L}$.

Therefore, a basic formulation of a geometric yield analysis is to compute the worst-case statistical parameter vector among all parameter vectors which are outside or on the border of the acceptance region partition, if the nominal parameter vector satisfies the considered performance-specification feature (i.e. is inside the acceptance region partition), or which are inside or on the border of the acceptance region partition, if the nominal parameter vector violates the considered performance-specification feature (i.e. is outside the acceptance region partition):

$$\mathbf{x}_{s,0} \in A_{s,L/U} \; : \quad \max_{\mathbf{x}_s} \mathrm{pdf}_N(\mathbf{x}_s) \quad \text{s.t.} \quad \mathbf{x}_s \in \bar{A}_{s,L/U} \tag{264}$$

$$\mathbf{x}_{s,0} \in \bar{A}_{s,L/U} \; : \quad \max_{\mathbf{x}_s} \mathrm{pdf}_N(\mathbf{x}_s) \quad \text{s.t.} \quad \mathbf{x}_s \in A_{s,L/U} \tag{265}$$

Generally, (264) and (265) may have several solutions. In practice, a unique solution as illustrated in Figure 52 can be observed for most performance features. Exceptions are for instance mismatch-sensitive performance features for which (264) and (265) often lead to two worst-case parameter vectors for each mismatch-producing transistor pair, which are nearly symmetrical on both sides of the nominal parameter vector. This special case requires special solution algorithms.

Due to the presence of range-parameter tolerances, the constraints in (264) and (265) have to be worked out in more detail. According to (123) and (124), the parameter acceptance region partition of a performance-specification feature is defined as the set of those statistical parameter vectors, for which all range-parameter vectors satisfy the corresponding bound. For a lower performance-feature bound, this means that even the smallest performance-feature value that is obtained over all range-parameter vectors in their tolerance region must be greater than the lower performance-feature bound. For an upper performance-feature bound, it means that even the largest performance-feature value that is obtained over all range-parameter vectors in their tolerance region must be less than the upper performance-feature bound:

$$f \geq f_L : \qquad \mathbf{x}_s \in A_{s,L} \;\Leftrightarrow\; \min_{\mathbf{x}_r \in T_r} f(\mathbf{x}_s, \mathbf{x}_r) \geq f_L \tag{266}$$

$$f \leq f_U : \qquad \mathbf{x}_s \in A_{s,U} \;\Leftrightarrow\; \max_{\mathbf{x}_r \in T_r} f(\mathbf{x}_s, \mathbf{x}_r) \leq f_U \tag{267}$$

The parameter non-acceptance region partition of a performance-specification feature is complementary to the acceptance region and was defined as the set of those statistical parameter vectors, for which a range-parameter vector exists that violates the corresponding bound (131) and (132). For a lower performance-feature bound, this means that already the smallest performance-feature value that is obtained over all range-parameter vectors in their tolerance region is less than the lower performance-feature bound. For an performance-feature upper bound, it means that already the largest performance-feature value that is obtained over all range-parameter vectors in their tolerance region is greater than the upper performance-feature bound:

$$f \geq f_L : \qquad \mathbf{x}_s \in \bar{A}_{s,L} \;\Leftrightarrow\; \min_{\mathbf{x}_r \in T_r} f(\mathbf{x}_s, \mathbf{x}_r) < f_L \tag{268}$$

$$f \leq f_U : \qquad \mathbf{x}_s \in \bar{A}_{s,U} \;\Leftrightarrow\; \max_{\mathbf{x}_r \in T_r} f(\mathbf{x}_s, \mathbf{x}_r) > f_U \tag{269}$$

(266)–(269) are inserted into (264) and (265) to produce the problem formulation of geometric yield analysis for the four cases of a lower/upper bound that is satisfied/violated at the nominal statistical parameter vector. At the same time, we replace the maximization of the probability density function by the equivalent problem of minimizing β, which determines the equidensity contour according to (24).

$$f \geq f_L \text{ and } \mathbf{x}_{s,0} \in A_{s,L} : \quad \min_{\mathbf{x}_s, \mathbf{x}_r} \beta^2(\mathbf{x}_s) \;\text{ s.t. }\; \min_{\mathbf{x}_r \in T_r} f(\mathbf{x}_s, \mathbf{x}_r) \leq f_L \tag{270}$$

$$f \geq f_L \text{ and } \mathbf{x}_{s,0} \in \bar{A}_{s,L} : \quad \min_{\mathbf{x}_s, \mathbf{x}_r} \beta^2(\mathbf{x}_s) \;\text{ s.t. }\; \min_{\mathbf{x}_r \in T_r} f(\mathbf{x}_s, \mathbf{x}_r) \geq f_L \tag{271}$$

$$f \leq f_U \text{ and } \mathbf{x}_{s,0} \in A_{s,U} : \quad \min_{\mathbf{x}_s, \mathbf{x}_r} \beta^2(\mathbf{x}_s) \;\text{ s.t. }\; \max_{\mathbf{x}_r \in T_r} f(\mathbf{x}_s, \mathbf{x}_r) \geq f_U \tag{272}$$

$$f \leq f_U \text{ and } \mathbf{x}_{s,0} \in \bar{A}_{s,U} : \quad \min_{\mathbf{x}_s, \mathbf{x}_r} \beta^2(\mathbf{x}_s) \;\text{ s.t. }\; \max_{\mathbf{x}_r \in T_r} f(\mathbf{x}_s, \mathbf{x}_r) \leq f_U \tag{273}$$

$$\beta^2(\mathbf{x}_s) = (\mathbf{x}_s - \mathbf{x}_{s,0})^T \cdot \mathbf{C}^{-1} \cdot (\mathbf{x}_s - \mathbf{x}_{s,0}) \tag{274}$$

β represents the distance of a statistical parameter vector from the nominal parameter vector, measured as a weighted l_2-norm according to the equidensity contours of pdf_N. The solution of (270)–(273) leads to worst-case distances $\beta_{WL/U}$, which are the basis of a geometric approximation of the yield partitions and yields analogous to Sections 5.5 and 6.2.4.

(270)–(273) present the problem formulation of geometric yield analysis in the form of two nested nonlinear optimization problems. The outer optimization problem formulates the computation of the statistical parameter vector that can be seen from the nominal parameter vector on the other side of the border of the acceptance region partition and that has minimum weighted distance. The objective function is quadratic in the statistical parameters.

The inner optimization problem considers the range parameters that determine the border of the acceptance region in the space of statistical parameters. It is a single constraint with a nonlinear objective function. In what follows we assume the usual box constraints for the range parameters (8). Then, the inner optimization problem corresponds to a classical worst-case analysis as described in Section 5.1. The difference to a classical worst-case analysis is the nonlinear performance function in the constraints of (270)–(273).

An analytical solution of problems (270)–(273) cannot be formulated. The solution is computed based on numerical optimization, for instance with a deterministic approach based on Sequential Quadratic Programming. In the following, we will prepare a solution approach and properties of the solution.

6.3.2 Lagrangian Function

The Lagrangian functions of (270)–(273) will be formulated according to Appendix C. As there are two encapsulated optimization problems, we will formulate two encapsulated Lagrangian functions.

In (270)–(273), the inner worst-case analysis problem minimizes the performance feature if a lower bound is specified, and maximizes the performance feature if an upper bound is specified. This leads to two Lagrangian functions for the inner optimization problems of (270)–(273):

$$\begin{aligned}\mathcal{L}_{I,L}(\mathbf{x}_s, \mathbf{x}_r, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) &= f(\mathbf{x}_s, \mathbf{x}_r) \\ &\quad -\boldsymbol{\lambda}_1^T \cdot (\mathbf{x}_r - \mathbf{x}_{r,L}) - \boldsymbol{\lambda}_2^T \cdot (\mathbf{x}_{r,U} - \mathbf{x}_r) \qquad (275)\\ \mathcal{L}_{I,U}(\mathbf{x}_s, \mathbf{x}_r, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) &= -f(\mathbf{x}_s, \mathbf{x}_r) \\ &\quad -\boldsymbol{\lambda}_1^T \cdot (\mathbf{x}_r - \mathbf{x}_{r,L}) - \boldsymbol{\lambda}_2^T \cdot (\mathbf{x}_{r,U} - \mathbf{x}_r) \qquad (276)\end{aligned}$$

As mentioned, we have assumed box constraints for the range parameters (8). (275) applies in (270) and (271), (276) applies in (272) and (273). The negative sign of the performance function in (276) results from replacing "$\max f$" in (272) and (273) by "$-\min -f$." The Lagrangian functions of (270)–(273) can now be formulated as follows:

$$f \geq f_L \text{ and } \mathbf{x}_{s,0} \in A_{s,L}: \quad \mathcal{L}(\mathbf{x}_s, \mathbf{x}_r, \lambda, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) = \beta^2(\mathbf{x}_s) - \lambda \cdot (f_L - \mathcal{L}_{I,L}(\mathbf{x}_s, \mathbf{x}_r, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2)) \tag{277}$$

$$f \geq f_L \text{ and } \mathbf{x}_{s,0} \in \bar{A}_{s,L}: \quad \mathcal{L}(\mathbf{x}_s, \mathbf{x}_r, \lambda, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) = \beta^2(\mathbf{x}_s) + \lambda \cdot (f_L - \mathcal{L}_{I,L}(\mathbf{x}_s, \mathbf{x}_r, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2)) \tag{278}$$

$$f \leq f_U \text{ and } \mathbf{x}_{s,0} \in A_{s,U}: \quad \mathcal{L}(\mathbf{x}_s, \mathbf{x}_r, \lambda, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) = \beta^2(\mathbf{x}_s) + \lambda \cdot (f_U + \mathcal{L}_{I,U}(\mathbf{x}_s, \mathbf{x}_r, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2)) \tag{279}$$

$$f \leq f_U \text{ and } \mathbf{x}_{s,0} \in \bar{A}_{s,U}: \quad \mathcal{L}(\mathbf{x}_s, \mathbf{x}_r, \lambda, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) = \beta^2(\mathbf{x}_s) - \lambda \cdot (f_U + \mathcal{L}_{I,U}(\mathbf{x}_s, \mathbf{x}_r, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2)) \tag{280}$$

6.3.3 First-Order Optimality Condition

We will formulate the optimality conditions of (270) and (277) respectively. The other three cases can be formulated analogously.

The first-order optimality condition describing a stationary point of (277) and a solution of (270), $\mathbf{x}_{s,WL}$, $\mathbf{x}_{r,WL}$, λ_{WL}, $\boldsymbol{\lambda}_{1,WL}$, $\boldsymbol{\lambda}_{2,WL}$ and $\beta_{WL} = \beta(\mathbf{x}_{s,WL})$ is:

$f \geq f_L$ and $\mathbf{x}_{s,0} \in A_{s,L}$:

$$2\mathbf{C}^{-1} \cdot (\mathbf{x}_{s,WL} - \mathbf{x}_{s,0}) + \lambda_{WL} \cdot \underbrace{\nabla \mathcal{L}_{I,L}(\mathbf{x}_{s,WL})}_{\nabla f(\mathbf{x}_{s,WL})} = \mathbf{0} \tag{281}$$

$$\lambda_{WL} \cdot (f_L - \underbrace{\mathcal{L}_{I,L}(\mathbf{x}_{s,WL}, \mathbf{x}_{r,WL}, \boldsymbol{\lambda}_{1,WL}, \boldsymbol{\lambda}_{2,WL})}_{f(\mathbf{x}_{s,WL}, \mathbf{x}_{r,WL})}) = 0 \tag{282}$$

$$\min_{\mathbf{x}_r \in T_r} f(\mathbf{x}_{s,WL}, \mathbf{x}_r) \leq f_L \tag{283}$$

$$\lambda_{WL} \cdot \overbrace{(\nabla f(\mathbf{x}_{r,WL}) - \boldsymbol{\lambda}_{1,WL} + \boldsymbol{\lambda}_{2,WL})}^{\nabla \mathcal{L}_{I,L}(\mathbf{x}_{r,WL})} = \mathbf{0} \tag{284}$$

$$k = 1, \ldots, n_{xr}: \quad \lambda_{1,WL,k} \cdot (x_{r,WL,k} - x_{r,L,k}) = 0 \tag{285}$$

$$k = 1, \ldots, n_{xr}: \quad \lambda_{2,WL,k} \cdot (x_{r,U,k} - x_{r,WL,k}) = 0 \tag{286}$$

$$\mathbf{x}_{r,L} \leq \mathbf{x}_{r,WL} \leq \mathbf{x}_{r,U} \tag{287}$$

(281) results from the condition that $\nabla\mathcal{L}(\mathbf{x}_s) = \mathbf{0}$ must hold for a stationary point of the outer optimization problem. (284) results from the condition that $\nabla\mathcal{L}(\mathbf{x}_r) = \mathbf{0}$ must hold for a stationary point of the inner optimization problem.

(282) represents the complementarity condition of the outer optimization problem, (285) and (286) represent the complementarity condition of the inner optimization problem.

(283) is the constraint of the outer optimization problem, (287) is the constraint of the inner optimization problem.

$\nabla f(\mathbf{x}_{s,WL}) = \frac{\partial f}{\partial \mathbf{x}_s}\Big|_{\mathbf{x}_{s,WL},\mathbf{x}_{r,WL}}$ and $\nabla f(\mathbf{x}_{r,WL}) = \frac{\partial f}{\partial \mathbf{x}_r}\Big|_{\mathbf{x}_{s,WL},\mathbf{x}_{r,WL}}$ hold.

Note that (281) and (282) are very similar to the first-oder optimality condition of the general worst-case analysis (190) and (191). And that (284)–(286) have nearly the same form as the first-oder optimality condition of the classical worst-case analysis (153) and (155).

We can verify by contradiction that the constraint (283) is active at the solution, i.e. $f(\mathbf{x}_{s,WL}, \mathbf{x}_{r,WL}) = f_L$. If this constraint would be inactive at the solution, then $\lambda_{WL} = 0$. Using (281), then $\mathbf{x}_{s,WL} = \mathbf{x}_{s,0}$ would be the solution. It follows that $f(\mathbf{x}_{s,WL}, \mathbf{x}_{r,WL}) = f(\mathbf{x}_{s,0}, \mathbf{x}_{r,WL}) < f_L$ would hold, which is a contradiction to the initial situation that the nominal statistical parameter vector is satisfying the performance-feature bound. Therefore, the constraint (283) is active at the solution in all cases (277)–(280), i.e.:

$$\lambda_{WL/U} \;>\; 0 \tag{288}$$

$$f(\mathbf{x}_{s,WL/U}, \mathbf{x}_{r,WL/U}) \;=\; f_{L/U} \tag{289}$$

6.3.4 Second-Order Optimality Condition

The second-order optimality condition requires the Hessian matrix of the Lagrange function $\nabla^2\mathcal{L}(\mathbf{x}_{s,WL}, \mathbf{x}_{r,WL})$. $\nabla^2\mathcal{L}(\mathbf{x}_{s,WL}, \mathbf{x}_{r,WL})$ of (270) and (277) respectively in the case $f \geq f_L$ and $\mathbf{x}_{s,0} \in A_{s,L}$ is:

$$\nabla^2\mathcal{L}(\mathbf{x}_{s,WL}, \mathbf{x}_{r,WL}) = \begin{bmatrix} \frac{\partial^2\mathcal{L}}{\partial \mathbf{x}_s^2} & \frac{\partial^2\mathcal{L}}{\partial \mathbf{x}_s \partial \mathbf{x}_r} \\ \frac{\partial^2\mathcal{L}}{\partial \mathbf{x}_s \partial \mathbf{x}_r} & \frac{\partial^2\mathcal{L}}{\partial \mathbf{x}_r^2} \end{bmatrix}_{\mathbf{x}_{s,WL},\mathbf{x}_{r,WL}}$$

$$= \begin{bmatrix} 2\mathbf{C}^{-1} + \lambda_{WL} \cdot \nabla^2 f(\mathbf{x}_{s,WL}) & \lambda_{WL} \cdot \frac{\partial^2 f}{\partial \mathbf{x}_s \partial \mathbf{x}_r}\Big|_{\mathbf{x}_{s,WL},\mathbf{x}_{r,WL}} \\ \lambda_{WL} \cdot \frac{\partial^2 f}{\partial \mathbf{x}_s \partial \mathbf{x}_r}\Big|_{\mathbf{x}_{s,WL},\mathbf{x}_{r,WL}} & \lambda_{WL} \cdot \nabla^2 f(\mathbf{x}_{r,WL}) \end{bmatrix} \tag{290}$$

Nonnegative curvature of $\frac{\partial^2\mathcal{L}}{\partial \mathbf{x}_r^2}$ at the worst-case parameter vector for unconstrained directions describes that the performance function is bounded below with respect to the range parameters. This is required for the existence of a border of the acceptance region A_s as defined in (123).

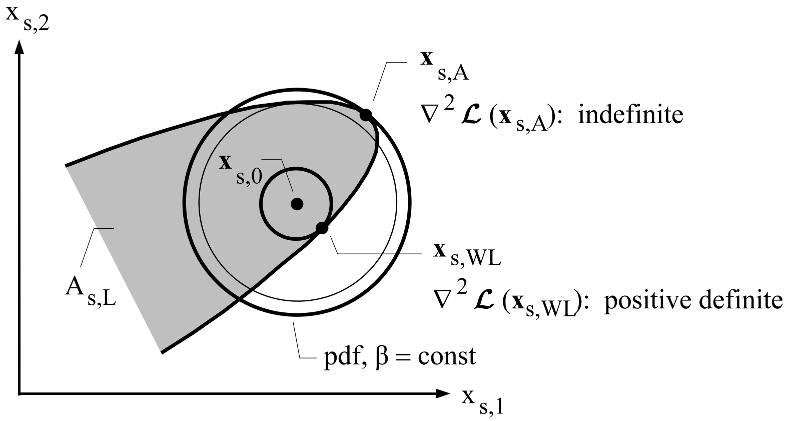

$x_{s,2}$
$\mathbf{x}_{s,A}$
$\nabla^2 \mathcal{L}(\mathbf{x}_{s,A})$: indefinite
$\mathbf{x}_{s,0}$
$\mathbf{x}_{s,WL}$
$\nabla^2 \mathcal{L}(\mathbf{x}_{s,WL})$: positive definite
$A_{s,L}$
pdf, β = const
$x_{s,1}$

Figure 53. Two stationary points $\mathbf{x}_{s,A}$ and $\mathbf{x}_{s,W}$ of (277), which satisfy the first-order optimality condition. Only $\mathbf{x}_{s,W}$ satisfies the second-order optimality condition and is therefore a solution of (270).

Nonnegative curvature of $\frac{\partial^2 \mathcal{L}}{\partial \mathbf{x}_s^2}$ at the worst-case parameter vector for unconstrained directions corresponds to the relation between the curvature of the tolerance ellipsoid of the statistical distribution and the curvature of the border of the acceptance region. For $\mathbf{x}_{s,WL}$ to be a minimum of (270), the curvature of the tolerance ellipsoid has to be stronger than the curvature of the border of the acceptance region as in the example in Figure 52.

Figure 53 shows another example. Both parameter vectors $\mathbf{x}_{s,A}$ and $\mathbf{x}_{s,WL}$ satisfy the first-order optimality condition, but only $\mathbf{x}_{s,WL}$ satisfies the second-order optimality condition as well. In $\mathbf{x}_{s,A}$, the curvature of the border of the acceptance region is stronger than the curvature of the corresponding tolerance region. Therefore, this tolerance region is not a subset of the acceptance region, and there are points inside this tolerance region but outside of the acceptance region for which the probability density value or the corresponding β value is larger than at $\mathbf{x}_{s,A}$. Therefore, $\mathbf{x}_{s,A}$ is not a solution of (270).

Note that there is a third stationary point of (277) in Figure 53, where the thin circle touches the border of the acceptance region. This point satisfies the second-order optimality condition and is a local minimum of (270). A similar situation will occur for locally varying mismatch-producing parameters. From a deterministic optimizer starting from $\mathbf{x}_{s,0}$ can be expected that it will find the global minimum $\mathbf{x}_{s,WL}$, as $\mathbf{x}_{s,WL}$ is closer to $\mathbf{x}_{s,0}$ due to the problem formulation. But as there is no guaranty for that and as we might be interested to know all local minima, suitable measures have to be taken.

6.3.5 Worst-Case Range-Parameter Vector

We can formulate the worst-case range-parameter vector $\mathbf{x}_{r,WL}$ based on (284)–(287). Due to (288), λ_{WL} can be eliminated from (284). (284)–(286) then have the same form as (153)–(155) of a classical worst-case analysis.

If the worst-case range-parameter vector is in a corner of the tolerance region T_r, it can therefore be formulated by replacing the performance gradients at the nominal parameter vector with the performance gradients at the worst-case parameter vector in the corresponding equations (158) and (159) from the worst-case analysis in Section 5.1. The components of the worst-case parameter vector $\mathbf{x}_{r,WL/U} = [\ldots x_{r,WL/U,k} \ldots]^T$ then are determined by:

$$x_{r,WL,k} = \begin{cases} x_{r,L,k}, & \nabla f(x_{r,WL,k}) > 0 \\ x_{r,U,k}, & \nabla f(x_{r,WL,k}) < 0 \end{cases} \tag{291}$$

$$x_{r,WU,k} = \begin{cases} x_{r,U,k}, & \nabla f(x_{r,WU,k}) > 0 \\ x_{r,L,k}, & \nabla f(x_{r,WU,k}) < 0 \end{cases} \tag{292}$$

(291) and (292) can be applied to save computational cost of the iterative solution of (270)–(273). Towards that, the worst-case range-parameter vector is initialized based on (291), (292) and on the performance-feature gradient at the nominal parameter vector. This initial worst-case range-parameter vector is not included in the Sequential-Quadratic-Programming solution of (270)–(273). A monitoring of the gradients with regard to the range parameters is applied instead to iteratively update the worst-case range-parameter vector. It has turned out in practice that this partitioning of the solution of (270)–(273) leads to a saving in computational cost due to the often plain behavior of the performance function with respect to range parameters.

6.3.6 Worst-Case Statistical Parameter Vector

From (281) follows in the case $f \geq f_L$ and $\mathbf{x}_{s,0} \in A_{s,L}$:

$$\mathbf{x}_{s,WL} - \mathbf{x}_{s,0} \quad = \quad -\frac{\lambda_{WL}}{2} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL}) \tag{293}$$

Inserting (293) in (274) leads to:

$$\frac{\lambda_{WL}^2}{4} \cdot \nabla f(\mathbf{x}_{s,WL})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL}) = \beta_{WL}^2$$

$$\frac{\lambda_{WL}}{2} = \frac{\beta_{WL}}{\sqrt{\nabla f(\mathbf{x}_{s,WL})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL})}} \tag{294}$$

Inserting (294) in (293) yields the formulation of the statistical worst-case parameter vector for the case of a lower performance-feature bound that is satisfied at the nominal parameter vector. The other cases are obtained analogously, starting from the Lagrangian functions (277)–(280).

$$f \geq f_L \text{ and } \mathbf{x}_{s,0} \in A_{s,L} \ , \ f \leq f_U \text{ and } \mathbf{x}_{s,0} \in \bar{A}_{s,U} : \tag{295}$$

$$\mathbf{x}_{s,WL/U} - \mathbf{x}_{s,0} = \frac{-\beta_{WL/U}}{\sqrt{\nabla f(\mathbf{x}_{s,WL/U})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U})}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U})$$

$$f \geq f_L \text{ and } \mathbf{x}_{s,0} \in \bar{A}_{s,L} \ , \ f \leq f_U \text{ and } \mathbf{x}_{s,0} \in A_{s,U} : \tag{296}$$

$$\mathbf{x}_{s,WL/U} - \mathbf{x}_{s,0} = \frac{+\beta_{WL/U}}{\sqrt{\nabla f(\mathbf{x}_{s,WL/U})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U})}} \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U})$$

Note that (295), (296), which describe the worst-case statistical parameter vectors of a geometric yield analysis, are identical to (194), (196), which describe the worst-case statistical parameter vectors of a general worst-case analysis.

6.3.7 Worst-Case Distance

The worst-case statistical parameter vector $\mathbf{x}_{s,WL/U}$ describes a tolerance ellipsoid according to (274):

$$\beta_{WL/U}^2 = \beta^2(\mathbf{x}_{s,WL/U}) = (\mathbf{x}_{s,WL/U} - \mathbf{x}_{s,0})^T \mathbf{C}^{-1} (\mathbf{x}_{s,WL/U} - \mathbf{x}_{s,0}) \tag{297}$$

The worst-case distance $\beta_{WL/U}^2$ determines the maximal tolerance ellipsoid of statistical parameters that touches the boundary of the parameter acceptance region for the considered performance-specification feature, i.e. performance-feature bound. In Figure 52, this tolerance ellipsoid is marked with an increased line thickness.

The linearization of the performance-feature function at the worst-case parameter vector is:

$$\bar{f}^{(WL/U)}(\mathbf{x}_s) = f_{L/U} + \nabla f(\mathbf{x}_{s,WL/U})^T \cdot (\mathbf{x}_s - \mathbf{x}_{s,WL/U}) \tag{298}$$

From (298) we obtain:

$$-\nabla f(\mathbf{x}_{s,WL/U})^T \cdot (\mathbf{x}_{s,WL/U} - \mathbf{x}_{s,0}) = \bar{f}^{(WL/U)}(\mathbf{x}_{s,0}) - f_{L/U} \tag{299}$$

Inserting (295) or (296) in (299) leads to

$$f \geq f_L \text{ and } \mathbf{x}_{s,0} \in A_{s,L} \ , \ f \leq f_U \text{ and } \mathbf{x}_{s,0} \in \bar{A}_{s,U} :$$

$$\beta_{WL/U} = \frac{\bar{f}^{(WL/U)}(\mathbf{x}_{s,0}) - f_{L/U}}{\sqrt{\nabla f(\mathbf{x}_{s,WL/U})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U})}} \tag{300}$$

$$= \frac{\nabla f(\mathbf{x}_{s,WL/U})^T \cdot (\mathbf{x}_{s,0} - \mathbf{x}_{s,WL/U})}{\sigma_{\bar{f}(WL/U)}}$$

$f \geq f_L$ and $\mathbf{x}_{s,0} \in \bar{A}_{s,L}$, $f \leq f_U$ and $\mathbf{x}_{s,0} \in A_{s,U}$:

$$\begin{aligned} \beta_{WL/U} &= \frac{f_{L/U} - \bar{f}^{(WL/U)}(\mathbf{x}_{s,0})}{\sqrt{\nabla f(\mathbf{x}_{s,WL/U})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U})}} \\ &= \frac{\nabla f(\mathbf{x}_{s,WL/U})^T \cdot (\mathbf{x}_{s,WL/U} - \mathbf{x}_{s,0})}{\sigma_{\bar{f}(WL/U)}} \end{aligned} \tag{301}$$

Note that (300) and (301), which describe the worst-case distance from a geometric yield analysis, are identical to the worst-case distance from a single-plane-bounded tolerance region (257), and are identical to the worst-case distance from a general worst-case analysis (200)–(203).

In all cases, a worst-case distance is defined as a multiple of a performance standard deviation $\sigma_{\bar{f}(WL)}$. Specifically, it is the standard deviation of the linearized performance at the worst-case parameter vector.

Moreover, (300) and (301) show that a change in the worst-case distance consists of two parts. On the one hand, the distance between the performance-feature bound and the nominal performance value has to be changed according to the nominator in (300) and (301). This corresponds to performance centering as described in Sections 2.4 and 2.5. On the other hand, the performance sensitivity with regard to the statistical parameters has to be changed according to the denominator in (300) and (301). The appropriate combination of both parts constitutes yield optimization/design centering. Note that we are aiming at increasing the worst-case distance if the nominal parameter vector is inside the parameter acceptance region partition, and that we are aiming at decreasing the worst-case distance if the nominal parameter vector is outside the parameter acceptance region partition.

6.3.8 Geometric Yield Partition

Based on the worst-case parameter vector according to (295), (296) and the performance-feature function linearized at the worst-case parameter vector (298), an approximation $A'_{s,L/U,i}$ of the parameter acceptance region partition (123), (124) is given (Figure 52).

This approximation corresponds to the single-plane-bounded tolerance class according to (258) and (259). Therefore, the worst-case distance either according to (297), using the worst-case parameter vector according to (295), (296), or according to (300), (301), using the performance-feature function linearized at the worst-case parameter vector (298), is directly related to the yield partition for one performance-specification feature:

$$\bar{Y}_{L/U,i} = \begin{cases} \displaystyle\int_{-\infty}^{\beta_{WL/U,i}} \frac{1}{\sqrt{2\pi}} \, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt \,, & f_i(\mathbf{x}_{s,0}) \in A_{s,L/U,i} \\ \displaystyle\int_{-\infty}^{-\,\beta_{WL/U,i}} \frac{1}{\sqrt{2\pi}} \, \mathrm{e}^{-\frac{1}{2}t^2} \cdot dt \,, & f_i(\mathbf{x}_{s,0}) \notin A_{s,L/U,i} \end{cases} \qquad (302)$$

$A_{s,L/U,i}$ are defined according to (123) and (124).

The corresponding yield values can be obtained from statistical tables or functions. Some values are given in Table 12. The accuracy of this geometric yield partition has been described for the general worst-case analysis in Section 5.5. In summary, the following can be stated about the accuracy of the geometric yield approximation with respect to one performance-specification feature:

- According to practical experience, the absolute yield error is around 1%–3%.
- The larger the actual yield value is, the smaller is the approximation error.
- The weaker the curvature of the true border of the specification-feature parameter acceptance region is compared to the curvature of the equidensity contours of the parameter distribution, the higher is the approximation quality.
- The worst-case distance inherently is suitable for yield values beyond 99.9%.
- Among all tangential planes of the parameter acceptance region partition, the one in the worst-case parameter vector provides a greatest lower bound or least upper bound on the yield partition approximation.

6.3.9 Geometric Yield

A geometric yield analysis is done for each individual performance-specification feature. As a result, for each individual performance-specification feature, a worst-case parameter vector, a worst-case distance, a performance-feature function linearization and an approximate yield partition value are obtained.

For an operational amplifier like that in Figure 3, Table 14 shows five performance features with an either lower or upper bound, the nominal performance feature values and the worst-case distances for each of the five performance-specification features, which result from a geometric yield analysis.

While it is not possible to judge whether a performance safety margin of $11dB$ for the gain is better or worse than a performance safety margin of $37MHz$ for the transit frequency, the worst-case distances computed through a geometric yield analysis immediately show that a worst-case distance of 2.5 for the gain means less robustness than a worst-case distance of 7.7 for the transit frequency.

Figure 54 shows the worst-case distances as a chart. The left axis in Figure 54

Table 14. Geometric yield analysis of an operational amplifier.

Performance feature	Specification feature	Nominal performance	Worst-case distance
Gain	$\geq 65dB$	$76dB$	2.5
Transit frequency	$\geq 30MHz$	$67MHz$	7.7
Phase margin	$\geq 60°$	$68°$	1.8
Slew rate	$\geq 32V/\mu s$	$67V/\mu s$	6.3
DC power	$\leq 3.5\mu W$	$2.6\mu W$	1.1
	$\bar{Y} = 82.9\%$		

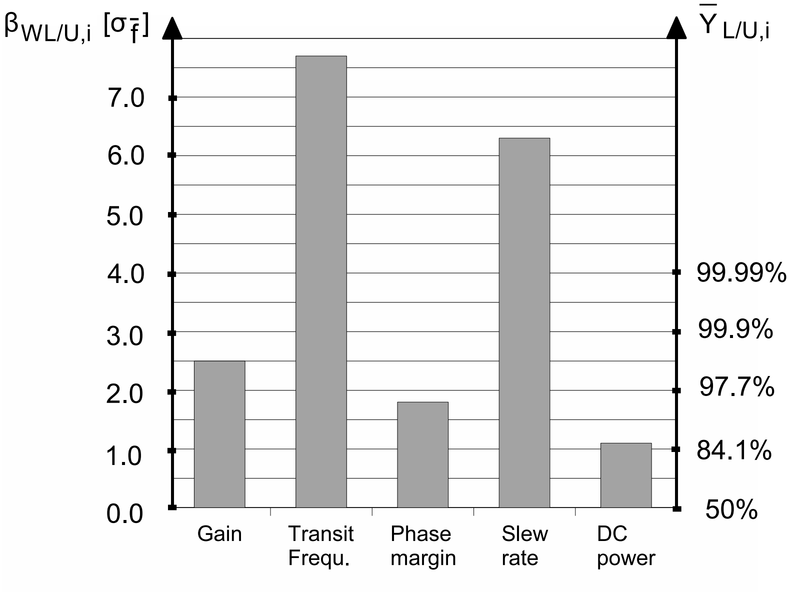


Figure 54. Worst-case distances and approximate yield values from a geometric yield analysis of the operational amplifier from Table 14.

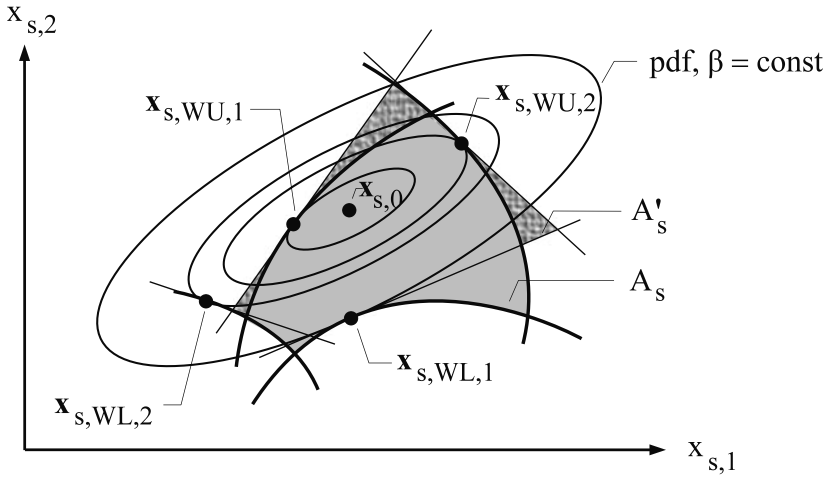


Figure 55. Parameter acceptance region A_s (gray area) originating from four performance-specification features, $f_1 \geq f_{L,1}$, $f_1 \leq f_{U,1}$, $f_2 \geq f_{L,2}$, $f_2 \leq f_{U,2}$ (Figure 32). A geometric yield analysis leads to four worst-case parameter vectors $\mathbf{x}_{WL,1}$, $\mathbf{x}_{WU,1}$, $\mathbf{x}_{WL,2}$, $\mathbf{x}_{WU,2}$ and four single-plane-bounded tolerance regions. The intersection of these single-plane-bounded tolerance regions forms the approximate parameter acceptance region A_s' (linen-pattern-filled area).

is scaled according to the worst-case distances. The unit is $\sigma_{\bar{f}}$, as a worst-case distance represents the performance safety margin as a multiple of the standard deviation of the linearized performance feature (298). The right axis in Figure 54 is scaled according to the yield partition value (302).

Figure 54 illustrates that the performance features gain, phase margin and DC power have relatively small worst-case distances leading to yield losses. The smallest worst-case distance with 1.1 is that of the DC power. Design centering obviously has to increase all these worst-case distances as much as possible, especially the smallest one.

The overall yield cannot be larger than the smallest yield value of the performance-specification features. The overall yield can be approximated by using the intersection of the parameter acceptance regions of the individual performance-specification features. This is illustrated in Figure 55. A Monte-Carlo analysis using the approximate parameter acceptance region A_s' can be performed at no additional simulation cost and is therefore very fast. The yield is approximated as given in the last row of Table 14. This value has been confirmed with a 2% accuracy by a Monte-Carlo analysis based on numerical simulation.

Practical experience shows that a geometric yield analysis requires $k' = 1 \ldots 7$ iterative steps of an SQP-based solution algorithm. In each iteration step, a sensitivity analysis has to be performed, whose complexity is proportional to the number of parameters n_{xs}. As sensitivity analyses have to be done for each performance-specification feature, the total number of analyses is proportional to the number of performance features n_f.

Overall we have for a geometric yield analysis:

$$\begin{array}{rcl} \text{Accuracy} & \lesssim & 3\% \\ \text{Complexity} & \sim & k \cdot n_f \cdot n_{xs} \end{array}$$

Compared to a Monte-Carlo analysis, the accuracy depends on the curvature of the performance function in the worst-case parameter vector.

The complexity depends on the number of parameters and the number of performance-specification features.

For problems with around 50 parameters, a whole yield optimization/design centering process based on geometric yield analyses can be done at the cost of one Monte-Carlo analysis.

6.3.10 General Worst-Case Analysis/Geometric Yield Analysis

In Sections 4.7 and 4.8, the input and output quantities of a worst-case analysis (see Figure 28 and (112)) and a yield analysis (see Figure 33 and (140)) have been discussed. In Sections 5.4 and 6.3, the general worst-case analysis and the geometric yield analysis have been formulated and analyzed.

Based on the probability density function of statistical parameters, on a tolerance region of range parameters and on a nominal parameter vector, the general worst-case analysis basically maps a worst-case distance onto worst-case performance values:

$$\text{General worst-case analysis: } \beta_W \quad \mapsto \quad f_{WL/U,i}\,,\ i = 1, \ldots, n_f \tag{303}$$

The worst-case distance corresponds to a minimum yield value based on the single-plane-bounded tolerance class (Section 6.2.4). The geometric yield analysis has the same basis as the general worst-case analysis and basically maps performance-specification features onto worst-case distances:

$$\text{Geometric yield analysis: } f_{WL/U,i} \quad \mapsto \quad \beta_{WL/U,i}\,,\ i = 1, \ldots, n_f \tag{304}$$

The obtained worst-case distances are transformed into performance-specification-feature yield values based on single-plane-bounded tolerance class (Section 6.2.4). Both the general worst-case analysis and the geometric yield analysis additionally compute worst-case statistical parameter vectors and worst-case range-parameter vectors. Obviously, the general worst-case analysis is the

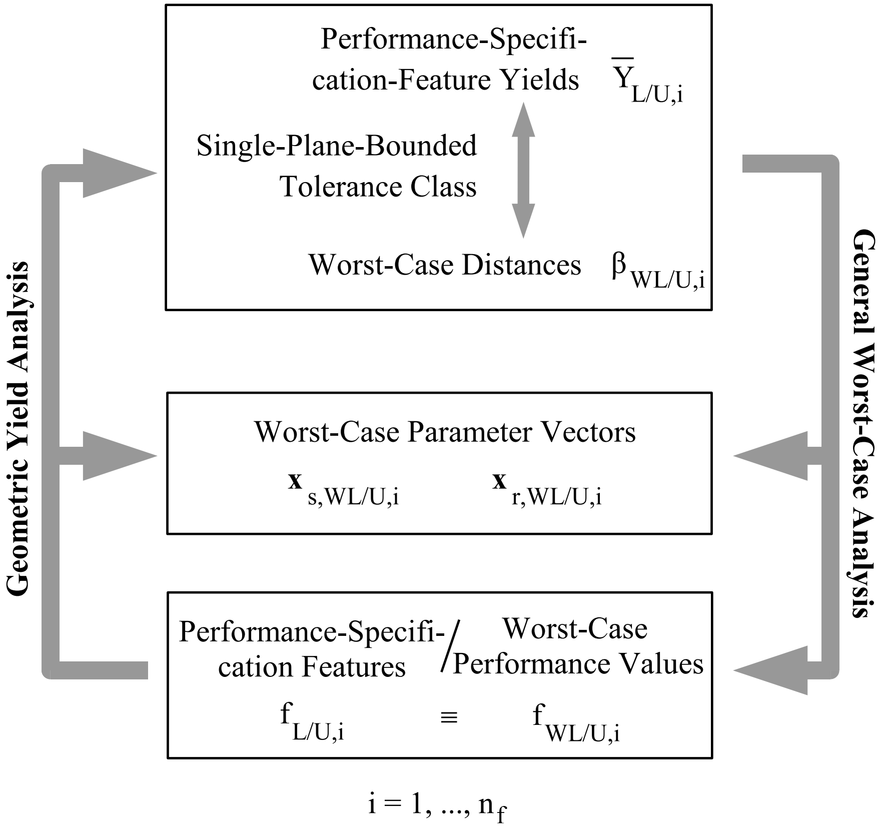

Performance-Specification-Feature Yields $\bar{Y}_{L/U,i}$
Single-Plane-Bounded Tolerance Class
Worst-Case Distances $\beta_{WL/U,i}$
Worst-Case Parameter Vectors
$\mathbf{x}_{s,WL/U,i}$
$\mathbf{x}_{r,WL/U,i}$
Performance-Specification Features / Worst-Case Performance Values
$f_{L/U,i} \equiv f_{WL/U,i}$
$i = 1, ..., n_f$
Geometric Yield Analysis
General Worst-Case Analysis

Figure 56. General worst-case analysis and geometric yield analysis as inverse mappings exchanging input and output.

inverse mapping of the geometric yield analysis. This mapping is bijective if the worst-case parameter vectors are unique. Figure 56 illustrates the inverse character of the two tasks, where the output worst-case performance values of a general worst-case analysis turn into input performance-specification features of a geometric yield analysis.

6.3.11 Approximate Geometric Yield Analysis

An iterative deterministic solution algorithm for a geometric yield is starting from a sensitivity analysis of the performance features with regard to the parameters at the nominal parameter vector.

An approximate geometric yield analysis could be performed by restriction to this first step. This compares to a general worst-case analysis, where the first step corresponds to a realistic worst-case analysis.

6.4 Exercise

- Given is the example of Section 5.6 with a single performance function of two parameters:

  $$f = x_{s,1} \cdot x_{s,2} \tag{305}$$

  f could be for instance the time constant of the RC circuit in Chapter 2. The nominal parameter vector is:

  $$\mathbf{x}_{s,0} = [\,1\ 1\,]^T \tag{306}$$

  The parameters are normally distributed with the covariance matrix:

  $$\mathbf{C} = \begin{bmatrix} 0.2^2 & 0 \\ 0 & 0.8^2 \end{bmatrix} \tag{307}$$

  Two performance-specification features are given:

  $$f \geq f_L \equiv 0.5 \tag{308}$$

  $$f \leq f_U \equiv 2.0 \tag{309}$$

  Perform a geometric yield analysis for the two performance-specification features. Apply the optimality conditions (Appendix C) to calculate a solution.

- Given is the following single performance function of two parameters:

  $$f = x_{s,1}^2 \cdot \frac{1}{4} x_{s,2}^2 \tag{310}$$

  The nominal parameter vector is:

  $$\mathbf{x}_{s,0} = [\,0\ 0\,]^T \tag{311}$$

  The parameters are normally distributed with the covariance matrix:

  $$\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{312}$$

  One performance-specification feature is given:

  $$f \leq f_U \equiv 1.0 \tag{313}$$

  Perform a geometric yield analysis. Apply the optimality conditions (Appendix C) to calculate a solution. Check the positive definiteness of $\nabla^2 \mathcal{L}(\mathbf{x}_{r,WL})$ and the second-order optimality condition (290) to verify the solution.

Chapter 7

YIELD OPTIMIZATION/DESIGN CENTERING

In this chapter, the problem formulation of yield optimization/design centering of Section 4.8.5 will be further developed. Two basic directions to approach yield optimization/design centering will be explained.

The first approach is based on a statistical yield analysis with a Monte-Carlo analysis described in Section 6.1. The gradient and the Hessian matrix of the statistically determined yield with regard to the nominal values of statistical parameters will be derived [8, 11, 10]. The statistical yield gradient and Hessian is calculated based on the results of a Monte-Carlo analysis and enable a Newton-type deterministic solution approach to statistical-yield optimization.

The second approach is based on a geometric yield analysis described in Section 6.3. The gradients of the worst-case distances, which result from a geometric yield analysis, with regard to any parameter will be derived [6]. Worst-case distances and their gradients lead to a geometric-yield optimization approach, which can be solved with multiple-objective optimization methods developed for nominal design.

7.1 Statistical-Yield Optimization

7.1.1 Acceptance-Truncated Distribution

Figures 18 and 46 illustrated that the probability density function of statistical parameters is truncated by the acceptance function (134). All those parts of the probability density function are cut away which correspond to parameter vectors that violate at least one of the performance-specification features according to (127), (128), or (131), (132). The resulting truncated probability density function pdf_δ is not a normal distribution:

$$\mathrm{pdf}_\delta(\mathbf{x}_s) = \frac{1}{Y} \cdot \delta(\mathbf{x}_s) \cdot \mathrm{pdf}(\mathbf{x}_s) \qquad (314)$$

The factor $\frac{1}{Y}$ makes pdf_δ satisfy (15):

$\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} \text{pdf}_\delta(\mathbf{x}_s)\cdot d\mathbf{x}_s = \frac{1}{Y}\cdot\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} \delta(\mathbf{x}_s)\cdot\text{pdf}(\mathbf{x}_s)\cdot d\mathbf{x}_s = \frac{1}{Y}\cdot Y = 1$

The mean value $\mathbf{x}_{s,0,\delta}$ and the covariance matrix $\mathbf{C}_\delta$ of the truncated probability density function pdf_δ can be formulated according to Appendix A:

$$\mathbf{x}_{s,0,\delta} = \underset{\text{pdf}_\delta}{\text{E}}\{\mathbf{x}_s\} = \frac{1}{Y}\cdot\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}\mathbf{x}_s\cdot\delta(\mathbf{x}_s)\cdot\text{pdf}(\mathbf{x}_s)\cdot d\mathbf{x}_s \tag{315}$$

$$\begin{aligned}\mathbf{C}_\delta &= \underset{\text{pdf}_\delta}{\text{E}}\{(\mathbf{x}_s-\mathbf{x}_{s,0,\delta})\cdot(\mathbf{x}_s-\mathbf{x}_{s,0,\delta})^T\}\\ &= \frac{1}{Y}\int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}(\mathbf{x}_s-\mathbf{x}_{s,0,\delta})(\mathbf{x}_s-\mathbf{x}_{s,0,\delta})^T\delta(\mathbf{x}_s)\text{pdf}(\mathbf{x}_s)d\mathbf{x}_s\end{aligned} \tag{316}$$

An estimator of the mean value $\widehat{\mathbf{x}}_{s,0,\delta}$ and an estimator of the covariance matrix $\widehat{\mathbf{C}}_\delta$ of the truncated probability density function pdf_δ can be formulated according to Appendix B:

$$\widehat{\mathbf{x}}_{s,0,\delta} = \frac{1}{n_{ok}}\sum_{\mu=1}^{n_{MC}}\delta(\mathbf{x}_s^{(\mu)})\cdot\mathbf{x}_s^{(\mu)} \tag{317}$$

$$\widehat{\mathbf{C}}_\delta = \frac{1}{n_{ok}-1}\sum_{\mu=1}^{n_{MC}}\delta(\mathbf{x}_s^{(\mu)})(\mathbf{x}_s^{(\mu)}-\widehat{\mathbf{x}}_{s,0,\delta})(\mathbf{x}_s^{(\mu)}-\widehat{\mathbf{x}}_{s,0,\delta})^T \tag{318}$$

$$n_{ok} = \sum_{\mu=1}^{n_{MC}}\delta(\mathbf{x}_s^{(\mu)}) = \widehat{Y}\cdot n_{MC}\approx Y\cdot n_{MC} \tag{319}$$

These estimators can be computed within a Monte-Carlo analysis according to Section 6.1.

7.1.2 Statistical Yield Gradient

The gradient of the statistically estimated yield with regard to the nominal statistical parameter vector $\nabla Y(\mathbf{x}_{s,0})$ can be calculated in the following way:

$$\nabla Y(\mathbf{x}_{s,0}) \overset{(137)}{=} \int_{-\infty}^{+\infty}\cdots\int_{-\infty}^{+\infty}\delta(\mathbf{x}_s)\cdot\nabla\text{pdf}_N(\mathbf{x}_{s,0})\cdot d\mathbf{x}_s \tag{320}$$

$$\overset{(A.1)}{=} \text{E}\left\{\frac{\delta(\mathbf{x}_s)}{\text{pdf}_N(\mathbf{x}_s)}\cdot\nabla\text{pdf}_N(\mathbf{x}_{s,0})\right\} \tag{321}$$

$$\overset{(23),(24)}{=} \int_{-\infty}^{+\infty}\ldots\int_{-\infty}^{+\infty}\delta(\mathbf{x}_s)\mathbf{C}^{-1}(\mathbf{x}_s-\mathbf{x}_{s,0})\mathrm{pdf}_N(\mathbf{x}_s)d\mathbf{x}_s \quad (322)$$

$$\begin{aligned} = \quad & \mathbf{C}^{-1}\cdot\left[\int_{-\infty}^{+\infty}\ldots\int_{-\infty}^{+\infty}\mathbf{x}_s\cdot\delta(\mathbf{x}_s)\cdot\mathrm{pdf}_N(\mathbf{x}_s)\cdot d\mathbf{x}_s\right.\\ & \left.-\mathbf{x}_{s,0}\cdot\int_{-\infty}^{+\infty}\ldots\int_{-\infty}^{+\infty}\delta(\mathbf{x}_s)\cdot\mathrm{pdf}_N(\mathbf{x}_s)\cdot d\mathbf{x}_s\right] \quad (323)\\ = \quad & \mathbf{C}^{-1}\cdot[Y\cdot\mathbf{x}_{s,0,\delta}-\mathbf{x}_{s,0}\cdot Y]\end{aligned}$$

$$\nabla Y(\mathbf{x}_{s,0}) = Y\cdot\mathbf{C}^{-1}\cdot(\mathbf{x}_{s,0,\delta}-\mathbf{x}_{s,0}) \quad (324)$$

$\mathbf{x}_{s,0,\delta}$ denotes the mean value of the truncated probability density function pdf_δ according to (315).

From (324), the first-order optimality condition for a yield maximum $Y^* = Y(\mathbf{x}^*_{s,0})$ follows immediately:

$$\mathbf{x}^*_{s,0} = \mathbf{x}^*_{s,0,\delta} \quad (325)$$

(325) says that the optimal yield is achieved when the mean value of the truncated probability density function equals that of the original probability density function.

The mean value of a probability density function can be interpreted as the center of gravity of the mass represented by the volume under the probability density function. The first-order optimality condition (325) can therefore be interpreted in the sense, that the truncation of the probability density function due to the performance specification does not change the center of gravity in the optimum. This is the motivation for the term design centering. In the optimum nominal statistical parameter vector $\mathbf{x}^*_{s,0,\delta}$, the design is centered with regard to the performance specification in the sense that the center of gravity of the probability density function is not affected by the truncations due to the performance specification. Design centering means to find a sizing which represents an equilibrium concerning the center of gravity of the manufactured and tested "mass."

Note that a centering of the performance-feature values between their bounds is not a centered design according to this interpretation. Note also that geometrically inscribing a maximum tolerance ellipsoid in the parameter acceptance region neither is a centered design according to this interpretation.

Figure 57 illustrates the situation before and after having reached the equilibrium concerning the centers of gravity of original and truncated probability density function for two statistical parameters. Those parts of the equidensity contours that belong to the truncated probability density function are drawn in bold.

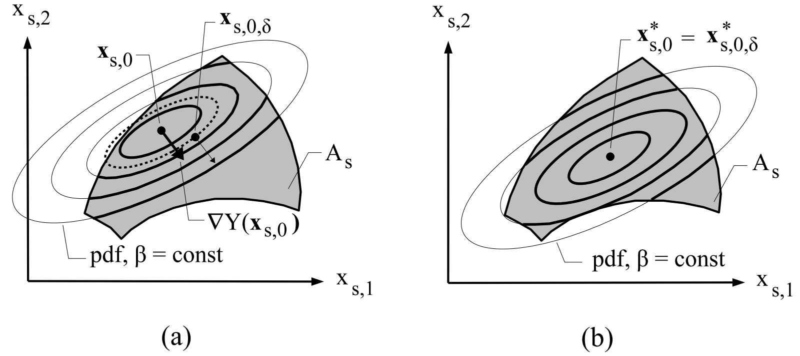


Figure 57. *(a)* Statistical-yield optimization before having reached the optimum. Center of gravity $\mathbf{x}_{s,0,\delta}$ of the probability density function truncated due to the performance specification (remaining parts drawn as bold line) differs from the center of gravity of the original probability density function. A nonzero yield gradient $\nabla Y(\mathbf{x}_{s,0})$ results. *(b)* After statistical-yield optimization having reached the optimum. Centers of gravity of original and truncated probability density function are identical.

The center of gravity $\mathbf{x}_{s,0,\delta}$ of the truncated probability density function is estimated by (317). It can be imagined as the point where the mass of the volume under the truncated probability density function can be balanced.

The gradient $\nabla Y(\mathbf{x}_{s,0})$ according to (324) has been drawn using the property that it is tangential to the equidensity contour through $\mathbf{x}_{s,0,\delta}$:

$$\frac{1}{2}\left.\frac{\partial \beta^2}{\partial \mathbf{x}_s}\right|_{\mathbf{x}_s=\mathbf{x}_{s,0,\delta}} \overset{(24)}{=} \mathbf{C}^{-1}\cdot(\mathbf{x}_{s,0,\delta}-\mathbf{x}_{s,0}) \tag{326}$$

7.1.3 Statistical Yield Hessian

In the same way as the gradient, the Hessian matrix of the yield with regard to the nominal statistical parameter vector $\nabla^2 Y(\mathbf{x}_{s,0})$ can be calculated:

$$\nabla^2 Y(\mathbf{x}_{s,0}) \overset{(137)}{=} \int_{-\infty}^{+\infty}\!\!\cdots\int_{-\infty}^{+\infty} \delta(\mathbf{x}_s)\cdot\nabla^2\mathrm{pdf}_N(\mathbf{x}_{s,0})\cdot d\mathbf{x}_s \tag{327}$$

The second-order derivative of the probability density function with regard to $\mathbf{x}_{s,0}$ can be calculated based on (23), (24) and using the first-order derivative in (322):

$$\begin{aligned}\nabla^2\mathrm{pdf}_N(\mathbf{x}_{s,0}) &= \mathbf{C}^{-1}(\mathbf{x}_s-\mathbf{x}_{s,0})\cdot\nabla\mathrm{pdf}_N(\mathbf{x}_{s,0}^T)-\mathbf{C}^{-1}\cdot\mathrm{pdf}_N(\mathbf{x}_s)\\ &= \left[\mathbf{C}^{-1}(\mathbf{x}_s-\mathbf{x}_{s,0})(\mathbf{x}_s-\mathbf{x}_{s,0})^T\mathbf{C}^{-1}-\mathbf{C}^{-1}\right]\mathrm{pdf}_N(\mathbf{x}_s)\\ &= \mathbf{C}^{-1}\left[(\mathbf{x}_s-\mathbf{x}_{s,0})(\mathbf{x}_s-\mathbf{x}_{s,0})^T-\mathbf{C}\right]\mathbf{C}^{-1}\mathrm{pdf}_N(\mathbf{x}_s) \quad (328)\end{aligned}$$

We insert (328) into (327) and extend the terms $\mathbf{x}_s - \mathbf{x}_{s,0}$ to $(\mathbf{x}_s - \mathbf{x}_{s,0,\delta}) + (\mathbf{x}_{s,0,\delta} - \mathbf{x}_{s,0})$:

$$\nabla^2 Y(\mathbf{x}_{s,0}) =$$
$$\mathbf{C}^{-1}\left[\int \cdots \int_{-\infty}^{+\infty} \delta(\mathbf{x}_s)[(\mathbf{x}_s - \mathbf{x}_{s,0,\delta}) + (\mathbf{x}_{s,0,\delta} - \mathbf{x}_{s,0})][\text{''}]^T \mathrm{pdf}_N(\mathbf{x}_s) d\mathbf{x}_s - Y\mathbf{C}\right]\mathbf{C}^{-1} \quad (329)$$

Using (315), (316) and (A.12)–(A.14), we obtain from (329) the Hessian matrix of the yield with regard to the nominal values of statistical parameters:

$$\nabla^2 Y(\mathbf{x}_{s,0}) = Y\mathbf{C}^{-1}\left[\mathbf{C}_\delta + (\mathbf{x}_{s,0,\delta} - \mathbf{x}_{s,0})(\mathbf{x}_{s,0,\delta} - \mathbf{x}_{s,0})^T - \mathbf{C}\right]\mathbf{C}^{-1} \quad (330)$$

$\mathbf{x}_{s,0,\delta}$ denotes the mean value of the truncated probability density function pdf_δ according to (315), and $\mathbf{C}_\delta$ denotes the covariance matrix of the truncated probability density function pdf_δ according to (316).

$\mathbf{x}_{s,0,\delta}$ and $\mathbf{C}_\delta$ can be estimated as part of a Monte-Carlo analysis using (317) and (318).

From (330) and the first-order optimality condition (325), the necessary second-order optimality condition for a yield maximum $Y^* = Y(\mathbf{x}^*_{s,0})$ follows immediately:

$$\mathbf{C}_\delta - \mathbf{C} \quad \text{is negative semidefinite} \quad (331)$$

(331) says that the yield is maximum if the variability of the truncated probability density function is smaller than that of the original probability density function. This expresses the property that the performance specification cuts away a part of the volume under the probability density function, and does this still in the optimum.

7.1.4 Solution Approach to Statistical-Yield Optimization

Based on the statistical gradient and Hessian matrix of the yield, (324) and (330), a quadratic model of the yield with respect to the nominal values of statistical parameters can be formulated:

$$\begin{aligned} Y^{(2)}(\mathbf{x}_{s,0}^{(next)}) &= Y(\mathbf{x}_{s,0}) + \nabla Y(\mathbf{x}_{s,0}) \cdot (\mathbf{x}_{s,0}^{(next)} - \mathbf{x}_{s,0}) \\ &\quad + \frac{1}{2}(\mathbf{x}_{s,0}^{(next)} - \mathbf{x}_{s,0})^T \cdot \nabla^2 Y(\mathbf{x}_{s,0}) \cdot (\mathbf{x}_{s,0}^{(next)} - \mathbf{x}_{s,0}) \quad (332) \end{aligned}$$

(332) has a stationary point according to Appendix C, where $\nabla Y^{(2)}(\mathbf{x}_{s,0}^{(next)}) \equiv \mathbf{0}$ holds:

$$\nabla Y^{(2)}(\mathbf{x}_{s,0}^{(next)}) \equiv \mathbf{0} \; : \; \nabla Y(\mathbf{x}_{s,0}) + \nabla^2 Y(\mathbf{x}_{s,0}) \cdot (\mathbf{x}_{s,0}^{(next)} - \mathbf{x}_{s,0}) = \mathbf{0} \quad (333)$$

The solution of this equation system produces a search direction

$$\mathbf{r} = \mathbf{x}_{s,0}^{(next)} - \mathbf{x}_{s,0} \quad (334)$$

for a Newton-type optimization approach. In the case that $\nabla^2 Y(\mathbf{x}_{s,0})$ is not positive semidefinite, special measures have to be taken. Such measures are:

- Ignore the Hessian matrix and use the gradient as search direction.
- Switch the signs of the negative eigenvalues of the Hessian.
- Bias the diagonal of the Hessian matrix in positive direction.

The quadratic model (332) reflects the behavior of the yield in a limited range of parameter values. To cope with the limited accuracy of the quadratic model, a line search along **r** is performed in a Newton-type optimization approach. It results in a new parameter vector:

$$\mathbf{x}_{s,0}^{(new)} = \mathbf{x}_{s,0} + \alpha \cdot \mathbf{r} \tag{335}$$

At the new parameter vector, another quadratic model according to (332) is computed, and the process described above restarts. This process is iteratively repeated until convergence is achieved.

For the line search, a quadratic model of the variance of the yield estimator with respect to the nominal values of statistical parameters can be formulated [8, 10]. An increase in the variance of the yield estimator goes together with the predicted yield improvement. At a certain point, the yield improvement that can be predicted with a required confidence reaches a maximum. This point can be used to determine the step length in (335).

7.1.5 Tolerance Assignment

For an optimization of yield by tuning of the covariance matrix, i.e. tolerance assignment (Section 4.8.6), the derivative of yield with respect to the covariance matrix, $\nabla Y(\mathbf{C})$ is required:

$$\nabla Y(\mathbf{C}) = \frac{1}{2} \cdot \nabla^2 Y(\mathbf{x}_{s,0}) \tag{336}$$

A solution approach to tolerance assignment based on (336) is presented in [11].

7.1.6 Deterministic Design Parameters

The yield optimization/design centering based on (324) and (330) works if the design parameters are at the same time statistical parameters. If design parameters have a deterministic character, a statistical estimation of yield by a Monte-Carlo analysis does not lead to yield derivatives.

A crude solution in that case would be a finite-difference approximation by repeated Monte-Carlo analyses:

$$\nabla Y(\mathbf{x}_d) \approx \frac{Y(\mathbf{x}_d') - Y(\mathbf{x}_d)}{\mathbf{x}_d' - \mathbf{x}_d} \tag{337}$$

Computing the statistical yield gradient with regard to deterministic design parameters according to (337) requires n_{xd} Monte-Carlo analyses. The resulting computational costs are prohibitive if numerical simulation is applied.

An alternative can be developed based on selecting a single statistical parameter $x_{s,k}$,

$$\mathbf{x}_s = \begin{bmatrix} x_{s,1} \\ \vdots \\ x_{s,k-1} \\ x_{s,k} \\ x_{s,k+1} \\ \vdots \\ x_{s,n_{xs}} \end{bmatrix} \in \mathbb{R}^{n_{xs}} \longrightarrow \mathbf{x}'_s = \begin{bmatrix} x_{s,1} \\ \vdots \\ x_{s,k-1} \\ x_{s,k+1} \\ \vdots \\ x_{s,n_{xs}} \end{bmatrix} \in \mathbb{R}^{n_{xs}-1}\ ,\ \ x_{s,k} \tag{338}$$

and formulating the yield (116) via the marginal distribution of the remaining statistical parameter $\mathbf{x}'_s$:

$$Y = \int_{-\infty}^{+\infty} \ldots \int_{-\infty}^{+\infty} \left[\int_{x_{s,k,L}(\mathbf{x}'_s)}^{x_{s,k,U}(\mathbf{x}'_s)} \mathrm{pdf}_N(x_{s,k}) \cdot dx_{s,k} \right] \mathrm{pdf}_N(\mathbf{x}'_s) \cdot d\mathbf{x}'_s \tag{339}$$

$$= \underset{\mathrm{pdf}_N(\mathbf{x}'_s)}{\mathrm{E}} \left\{ \mathrm{cdf}_N(x_{s,k,U}(\mathbf{x}'_s)) - \mathrm{cdf}_N(x_{s,k,L}(\mathbf{x}'_s)) \right\} \tag{340}$$

Here, we have assumed that the parameters have been transformed into standardized normally distributed random variables $\mathbf{x}_s$ with zero mean and unity covariance matrix, i.e. $\mathbf{x}_s \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. pdf_N and cdf_N are the probability density function and cumulative distribution function according to (16), (17) and (23).

$x_{s,k,U}$ and $x_{s,k,L}$ represent the borders of the parameter acceptance region A_s (122) projected onto the axis of parameter $x_{s,k}$

A statistical yield estimator based on (340) is determined using (B.1):

$$\widehat{Y} = \frac{1}{n_{MC}} \sum_{\mu=1}^{n_{MC}} (\mathrm{cdf}_N(x_{s,k,U}(\mathbf{x}_s^{\prime(\mu)})) - \mathrm{cdf}_N(x_{s,k,L}(\mathbf{x}_s^{\prime(\mu)}))) \tag{341}$$

$$\mathbf{x}_s^{\prime(\mu)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),\ \mu = 1, \ldots, n_{MC}$$

A higher accuracy is achieved if (341) is averaged over all available statistical parameters, $x_{s,k}$, $k = 1, \ldots, n_{xs}$:

$$\widehat{\widehat{Y}} = \frac{1}{n_{xs}} \sum_{k=1}^{n_{xs}} \frac{1}{n_{MC}} \sum_{\mu=1}^{n_{MC}} (\mathrm{cdf}_N(x_{s,k,U}(\mathbf{x}_s^{\prime(\mu)})) - \mathrm{cdf}_N(x_{s,k,L}(\mathbf{x}_s^{\prime(\mu)}))) \tag{342}$$

$$\mathbf{x}_s^{\prime(\mu)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),\ \mu = 1, \ldots, n_{MC}$$

(341) and (342) are evaluated based on a Monte-Carlo analysis. This requires the computation of $x_{s,k,U}(\mathbf{x}_s'^{(\mu)})$ and $x_{s,k,L}(\mathbf{x}_s'^{(\mu)})$ in each sample element $\mathbf{x}_s'^{(\mu)}$ by solving the following optimization problems, if the nominal statistical parameter vector is inside the acceptance region, $\mathbf{x}_{s,0} \in A_{sL/U}$:

$$\min_{x_{s,k},\mathbf{x}_r} (x_{s,k} - x_{s,k,0})^2 \quad \text{s.t.} \quad \left\{ \begin{array}{lcl} \min\limits_{\mathbf{x}_r \in T_r} f_i(\mathbf{x}_s'^{(\mu)}, x_{s,k}, \mathbf{x}_r) & \leq & f_{L,i} \\ \max\limits_{\mathbf{x}_r \in T_r} f_i(\mathbf{x}_s'^{(\mu)}, x_{s,k}, \mathbf{x}_r) & \geq & f_{U,i} \\ i = 1, \ldots, n_f & & \\ x_{s,k} \leq x_{s,k,0} & & \end{array} \right. \tag{343}$$

$$\min_{x_{s,k},\mathbf{x}_r} (x_{s,k} - x_{s,k,0})^2 \quad \text{s.t.} \quad \left\{ \begin{array}{lcl} \min\limits_{\mathbf{x}_r \in T_r} f_i(\mathbf{x}_s'^{(\mu)}, x_{s,k}, \mathbf{x}_r) & \leq & f_{L,i} \\ \max\limits_{\mathbf{x}_r \in T_r} f_i(\mathbf{x}_s'^{(\mu)}, x_{s,k}, \mathbf{x}_r) & \geq & f_{U,i} \\ i = 1, \ldots, n_f & & \\ x_{s,k} \geq x_{s,k,0} & & \end{array} \right. \tag{344}$$

(343) and (344) compare to the geometric yield analysis (264). The difference is that all performance-specification features are considered simultaneously and that only one parameter is considered in the objective function.

The solution of (343) and (344) becomes a line search along the parameter $x_{s,k}$ if the worst-case range-parameter vector can be predetermined as described in Section 6.3.5.

The statistical yield gradient with regard to deterministic design parameters is formulated starting from (340):

$$\begin{aligned} \nabla Y(\mathbf{x}_d) = & \underset{\text{pdf}_N(\mathbf{x}_s')}{\text{E}} \{\text{cdf}_N(x_{s,k,U}) \cdot \nabla x_{s,k,U}(\mathbf{x}_d) \\ & \qquad - \text{cdf}_N(x_{s,k,L}) \cdot \nabla x_{s,k,L}(\mathbf{x}_d)\} & (345) \\ = & \underset{\text{pdf}_N(\mathbf{x}_s')}{\text{E}} \left\{ \text{cdf}_N(x_{s,k,U}) \cdot \frac{\nabla f_i(\mathbf{x}_d)}{\nabla f_i(x_{s,k,U})} \right. \\ & \qquad \left. - \text{cdf}_N(x_{s,k,L}) \cdot \frac{\nabla f_j(\mathbf{x}_d)}{\nabla f_j(x_{s,k,L})} \right\} & (346) \end{aligned}$$

Here f_i and f_j are the performance features whose bounds are active in the solution of (343) and (344).

In addition to the n_{MC} simulations of a Monte-Carlo analysis, the computation of the statistical yield gradient with respect to deterministic design parameters requires at least $2n_{MC}$ line searches to solve (343) and (344) plus $2n_{MC}$ sensitivity analyses to solve (346).

The resulting simulation costs may still be prohibitive in practice. Methods based on statistical estimation of the yield gradient for deterministic design parameters like [43, 63] therefore fall back on response surface models.

7.2 Geometric-Yield Optimization

According to (302), a yield partition value is improved either by increasing the corresponding worst-case distance if the nominal parameter vector is inside its parameter acceptance region partition, or by decreasing the corresponding worst-case distance if the nominal parameter vector is outside its parameter acceptance region partition.

According to Figure 36, yield optimization/design centering by tuning of deterministic design parameters changes the shape of the parameter acceptance region in such a way that the truncation of the probability density function is minimized. According to Figure 37, yield optimization/design centering by tuning of the nominal values of the statistical parameters shifts the probability density function within the parameter acceptance region such that its truncation is minimized.

By partitioning of the parameter acceptance region according to individual performance-specification features as illustrated in Figure 32, a set of worst-case distances is obtained, each of which represents a partial amount of truncation due to the respective performance-specification feature.

Figure 52 indicates that the worst-case distances depend on both the shape of the parameter acceptance region and the nominal value of statistical parameters.

In the following, we will derive the worst-case-distance gradient, which has the same form and same computational cost for statistical and deterministic design parameters. This is in contrast to the statistical yield gradient, which is complicated for deterministic design parameters as has been shown in Section 7.1.6.

Based on worst-case distances, geometric-yield optimization will be formulated as a multiple-objective optimization problem. Two solution approaches for the geometric-yield optimization problem will be described.

7.2.1 Worst-Case-Distance Gradient

The starting point to calculate the worst-case-distance gradient is an extension of (300) and (301), which simultaneously shows the first-order dependence of the worst-case distance $\beta_{WL/U}$ on the mean values of statistical parameters $\mathbf{x}_{s,0}$, and on the (deterministic) design parameters $\mathbf{x}_d$.

Deterministic design parameters are included in the geometric yield analysis problem (270)-(273) by expanding the performance function $f(\mathbf{x}_s, \mathbf{x}_r)$ with a linear term with respect to the design parameters, which has been developed at

$\mathbf{x}_{d,\mu}$:

$$f(\mathbf{x}_s, \mathbf{x}_r) \rightarrow f(\mathbf{x}_s, \mathbf{x}_r) + \ \nabla f(\mathbf{x}_{d,\mu})^T \cdot (\mathbf{x}_d - \mathbf{x}_{d,\mu}) \qquad (347)$$

As this extension neglects second-order effects, it represents a constant for the inner optimization problem in (270)–(273). It can be passed on to the outer optimization problem and becomes a part of its constraint in that the bounds $f_{L/U}$ in (270)–(273) are extended by:

$$f_{L/U} \rightarrow f_{L/U} - \nabla f(\mathbf{x}_{d,\mu})^T \cdot (\mathbf{x}_d - \mathbf{x}_{d,\mu}) \qquad (348)$$

The replacement (348) proceeds to (300) and (301), which become:

$f \geq f_L$ and $\mathbf{x}_{s,0} \in A_{s,L}$, $f \leq f_U$ and $\mathbf{x}_{s,0} \in \bar{A}_{s,U}$: (349)

$$\beta_{WL/U} \ = \ \frac{\nabla f(\mathbf{x}_{s,WL})^T \cdot (\mathbf{x}_{s,0} - \mathbf{x}_{s,WL}) + \nabla f(\mathbf{x}_{d,\mu})^T \cdot (\mathbf{x}_d - \mathbf{x}_{d,\mu})}{\sqrt{\nabla f(\mathbf{x}_{s,WL})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL})}}$$

$f \geq f_L$ and $\mathbf{x}_{s,0} \in \bar{A}_{s,L}$, $f \leq f_U$ and $\mathbf{x}_{s,0} \in A_{s,U}$: (350)

$$\beta_{WL/U} \ = \ \frac{\nabla f(\mathbf{x}_{s,WL})^T \cdot (\mathbf{x}_{s,WL} - \mathbf{x}_{s,0}) - \nabla f(\mathbf{x}_{d,\mu})^T \cdot (\mathbf{x}_d - \mathbf{x}_{d,\mu})}{\sqrt{\nabla f(\mathbf{x}_{s,WL})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL})}}$$

From (349) and (350), the worst-case-distance gradients with respect to the mean values of statistical parameters $\mathbf{x}_{s,0}$ and with respect to (deterministic) design parameters $\mathbf{x}_d$ follow:

$f \geq f_L$ and $\mathbf{x}_{s,0} \in A_{s,L}$, $f \leq f_U$ and $\mathbf{x}_{s,0} \in \bar{A}_{s,U}$:

$$\nabla \beta_{WL/U}(\mathbf{x}_{s,0}) = \frac{+1}{\sqrt{\nabla f(\mathbf{x}_{s,WL/U})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U})}} \cdot \nabla f(\mathbf{x}_{s,WL/U}) \qquad (351)$$

$$\nabla \beta_{WL/U}(\mathbf{x}_d) = \frac{+1}{\sqrt{\nabla f(\mathbf{x}_{s,WL/U})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U})}} \cdot \nabla f(\mathbf{x}_{d,\mu}) \qquad (352)$$

$f \geq f_L$ and $\mathbf{x}_{s,0} \in \bar{A}_{s,L}$, $f \leq f_U$ and $\mathbf{x}_{s,0} \in A_{s,U}$:

$$\nabla \beta_{WL/U}(\mathbf{x}_{s,0}) = \frac{-1}{\sqrt{\nabla f(\mathbf{x}_{s,WL/U})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U})}} \cdot \nabla f(\mathbf{x}_{s,WL/U}) \qquad (353)$$

$$\nabla \beta_{WL/U}(\mathbf{x}_d) = \frac{-1}{\sqrt{\nabla f(\mathbf{x}_{s,WL/U})^T \cdot \mathbf{C} \cdot \nabla f(\mathbf{x}_{s,WL/U})}} \cdot \nabla f(\mathbf{x}_{d,\mu}) \qquad (354)$$

(351)–(354) show that the worst-case-distance gradient has the same form concerning statistical parameters and (deterministic) design parameters.

The worst-case-distance gradient corresponds to the performance gradient at the worst-case parameter vector. Its length is scaled according to the variance of the linearized performance (198), its direction depends on whether a lower or upper performance-feature bound is specified and whether it is satisfied or violated.

The equivalence of statistical parameters and deterministic design parameters in the worst-case distance (349) and (350) and in the worst-case-distance gradient (351)–(354) can also be interpreted using Figure 52.

The worst-case distance is increased by increasing the difference between the performance-feature value at the nominal statistical parameter vector and the performance-feature bound. In a linear performance model, this increase can be achieved by shifting any parameter of any kind. In Figure 52, which shows the subspace of parameters that are statistical, the performance gradient at the worst-case parameter vector shows the direction in which the nominal statistical parameter vector $\mathbf{x}_{s,0}$ has to be shifted for a steepest ascent in the worst-case distance according to (351). The same effect is achieved by a change in the (deterministic) design parameter vector $\mathbf{x}_d$ according to (352), which shifts the boundary of $A_{s,L}$ away from $\mathbf{x}_{s,0}$ according to (348).

Worst-Case-Distance Gradient by Lagrange Factor. From Appendix C follows that the sensitivity of the solution of (270)–(273) with respect to a perturbation ϵ in the constraint is determined by the Lagrange factor $\lambda_{W/L}$:

$$\nabla \beta^2_{WL/U}(\epsilon) = \lambda_{WL/U} \tag{355}$$

Inserting (348) into (277)–(280), we can find that ϵ is defined as:

$$f \geq f_L \text{ and } \mathbf{x}_{s,0} \in A_{s,L}, \quad f \leq f_U \text{ and } \mathbf{x}_{s,0} \in \bar{A}_{s,U} :$$

$$\epsilon \;=\; \nabla f(\mathbf{x}_{d,\mu})^T \cdot (\mathbf{x}_d - \mathbf{x}_{d,\mu}) \tag{356}$$

$$f \geq f_L \text{ and } \mathbf{x}_{s,0} \in \bar{A}_{s,L}, \quad f \leq f_U \text{ and } \mathbf{x}_{s,0} \in A_{s,U} :$$

$$\epsilon \;=\; -\nabla f(\mathbf{x}_{d,\mu})^T \cdot (\mathbf{x}_d - \mathbf{x}_{d,\mu}) \tag{357}$$

By the chain rule of differentiation and using (355), we obtain:

$$\begin{aligned}\nabla \beta_{WL/U}(\mathbf{x}_d) \;&=\; \frac{1}{\nabla \beta^2_{WL/U}(\beta_{WL/U})} \cdot \nabla \beta^2_{WL/U}(\epsilon) \cdot \nabla \epsilon(\mathbf{x}_d) \\ &=\; \frac{1}{2 \cdot \beta_{WL/U}} \cdot \lambda_{WL/U} \cdot \nabla \epsilon(\mathbf{x}_d)\end{aligned} \tag{358}$$

With (294) and based on differentiating (356) and (357) with respect to $\mathbf{x}_d$, we obtain (352) and (354).

7.2.2 Solution Approaches to Geometric-Yield Optimization

According to (302) and Section 6.3.8, each worst-case distance corresponds to an approximate yield partition value for the respective performance-specification feature if single-plane bounded tolerance classes (Section 6.2.4) are applied. Each worst-case distance thus represents a partition of the overall yield as illustrated in Figure 54. Yield optimization/design centering is achieved by maximizing those worst-case distances as much as possible, for which the nominal statistical parameter vector is inside the parameter acceptance region partition. If the nominal statistical parameter vector is outside the parameter acceptance region partition, then the corresponding worst-case distance is a measure for the degree of violation of the respective performance-specification feature. In this case, the worst-case distance has to be decreased. Once it reaches zero, the situation switches in that the nominal parameter vector gets inside the acceptance region partition and the worst-case distance becomes a robustness measure that shall be maximized.

Yield optimization/design centering in this way becomes a multiple-objective optimization problem according to (142) with worst-case distances as objectives:

$$\max_{\mathbf{x}_d} \left\{ \begin{array}{l} \alpha_i(\mathbf{x}_d) \cdot \beta_{WL/U,i}(\mathbf{x}_d) \\ i = 1, \ldots, n_f \end{array} \right\} \quad \text{s.t.} \quad \mathbf{c}(\mathbf{x}_d) \geq \mathbf{0} \tag{359}$$

$$\alpha_i = \left\{ \begin{array}{ll} +1, & \mathbf{x}_{s,0} \in A_{s,L/U,i}(\mathbf{x}_d) \\ -1, & \mathbf{x}_{s,0} \in \bar{A}_{s,L/U,i}(\mathbf{x}_d) \end{array} \right. \tag{360}$$

The sign α_i considers the two cases of the nominal statistical parameter vector lying inside or outside of the corresponding acceptance region partition.

Worst-case distance gradients (352), (354) are applied to solve (359).

Note that second-order derivatives are available for the statistical-yield optimization approach (332), while only first-order derivatives are available for the geometric-yield optimization approach (359). On the other hand, the statistical yield gradient with regard to deterministic parameters (346) involves impractical computational cost, while the worst-case distance gradient with regard to deterministic parameters (352), (354) is cheap.

For the example in Section 6.3.9, with the initial situation previously illustrated in Table 14 and Figure 54, yield optimization/design centering according to (359) leads to an optimal situation illustrated in the following Table 15 and Figure 58.

Compared to the initial situation after nominal design, the worst-case distances with smaller values have been maximized at the cost of those with larger

Table 15. Geometric-yield optimization of an operational amplifier from Section 6.3.9.

	After nominal design		After geometric-yield optimization	
Performance-specification feature	Nominal performance	Worst-case distance	Nominal performance	Worst-case distance
Gain $\geq 65dB$	$76dB$	2.5	$76dB$	4.2
Transit frequency $\geq 30MHz$	$67MHz$	7.7	$58MHz$	4.5
Phase margin $\geq 60°$	$68°$	1.8	$71°$	3.9
Slew rate $\geq 32V/\mu s$	$67V/\mu s$	6.3	$58V/\mu s$	3.9
DC power $\leq 3.5\mu W$	$2.6\mu W$	1.1	$2.3\mu W$	4.2
	$\bar{Y} = 82.9\%$		$\bar{Y} = 99.9\%$	

values by geometric-yield optimization. The smallest worst-case distances after geometric-yield optimization is 3.9 and means that this is nearly a four-sigma design. The smallest worst-case distance is obtained for the phase margin and the slew rate. The achieved value represents a Pareto point concerning the worst-case distances of these two performance features. None of them can be improved without worsening the other one. 3.9-sigma is the maximum robustness of speed and stability that can be achieved for the underlying process technology.

Note that the nominal performance value of the gain is the same before and after geometric-yield optimization, but the worst-case distance has increased from 2.5 to 4.2. According to (300) and (301), this must have been achieved by decreasing the sensitivity of gain with regard to the statistical parameters.

7.2.3 Least-Squares/Trust-Region Solution Approach

We will not consider the constraints $\mathbf{c}(\mathbf{x}_d) \geq \mathbf{0}$ for reasons of simplicity in the following.

(359) can be solved by formulating target values $\boldsymbol{\beta}_{W,target}$ for all worst-case distances, which have been collected in a vector $\boldsymbol{\beta}_{WL/U}$, and scalarizing the multiple-objective problem (106). The vector norm $\|.\|$ is meant to be the l_2-norm $\|.\|_2$ in the following.

$$\min_{\mathbf{x}_d} \|\boldsymbol{\beta}_{WL/U}(\mathbf{x}_d) - \boldsymbol{\beta}_{W,target}\|^2 \qquad (361)$$

$$\boldsymbol{\beta}_{WL/U} = [\ldots\ \alpha_i \cdot \beta_{WL/U,i}\ \ldots]^T$$

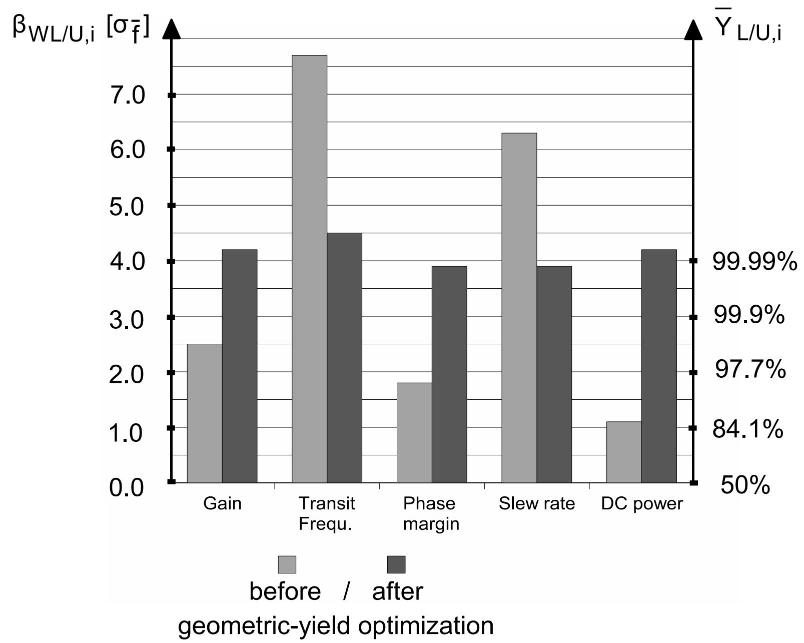


Figure 58. Worst-case distances and approximate yield partition values before and after geometric-yield optimization of the operational amplifier from Table 15.

(361) is a least-squares problem due to the objective function.

It is easy to formulate target values for the worst-case distances. For a three-sigma design for instance, all worst-case distances should be 3, for a six-sigma design, they should be 6.

Using the worst-case distance gradients (352), (354), a linear model of the objective function in (361) is established. First, the linear model of the vector of worst-case distances $\boldsymbol{\beta}_{WL/U}$ with regard to a search direction $\mathbf{r} = \mathbf{x}_d - \mathbf{x}_{d,\mu}$ starting from the current point $\mathbf{x}_{d,\mu}$ is formulated:

$$\begin{aligned} \bar{\boldsymbol{\beta}}_{WL/U}(\mathbf{r}) &= \boldsymbol{\beta}_{WL/U}(\mathbf{x}_{d,\mu}) + \underbrace{\nabla \boldsymbol{\beta}_{WL/U}(\mathbf{x}_{d,\mu})^T}_{\mathbf{J}} \cdot \mathbf{r} \qquad (362) \\ \text{with } \mathbf{r} &= \mathbf{x}_d - \mathbf{x}_{d,\mu} \end{aligned}$$

The matrix $\mathbf{J}$ in (362) contains all individual worst-case-distance gradients according to (352), (354) as rows.

Based on (362), the gradient of the linearized objective function in (361) is calculated:

$$\begin{aligned} \|\bar{\boldsymbol{\epsilon}}_F(\mathbf{r})\|^2 &= \|\bar{\boldsymbol{\beta}}_{WL/U}(\mathbf{r}) - \boldsymbol{\beta}_{W,target}\|^2 \\ &= \boldsymbol{\epsilon}_{F,0}^T \cdot \boldsymbol{\epsilon}_{F,0} + 2 \cdot \mathbf{r}^T \cdot \mathbf{J}^T \cdot \boldsymbol{\epsilon}_{F,0} + \mathbf{r}^T \cdot \mathbf{J}^T \cdot \mathbf{J} \cdot \mathbf{r} \qquad (363) \\ \text{with } \boldsymbol{\epsilon}_{F,0} &= \boldsymbol{\beta}_{WL/U}(\mathbf{x}_{d,\mu}) - \boldsymbol{\beta}_{W,target} \end{aligned}$$

Inserting (363) in (361) leads to a linear least-squares optimization problem,

$$\min_{\mathbf{r}} \|\bar{\boldsymbol{\epsilon}}_F(\mathbf{r})\|^2 \qquad (364)$$

which has a quadratic objective function due to the last term in (363). The stationary point of (364) is calculated as:

$$\nabla\|\bar{\boldsymbol{\epsilon}}_F\|^2(\mathbf{r}) \equiv \mathbf{0} : \qquad \mathbf{J}^T \cdot \mathbf{J} \cdot \mathbf{r} = -\mathbf{J} \cdot \boldsymbol{\epsilon}_{F,0} \qquad (365)$$

The solution of (365) is known as Gauss-Newton direction.

An improved computation of a search direction $\mathbf{r}$ considers the limited accuracy of the linearized model (362) by introducing a trust-region Δ that the search direction must not leave:

$$\min_{\mathbf{r}} \|\bar{\boldsymbol{\epsilon}}_F(\mathbf{r})\|^2 \ \text{ s.t. } \ \|\mathbf{r}\|^2 \leq \Delta^2 \qquad (366)$$

(366) represents a least-squares/trust-region approach to geometric-yield optimization. The Lagrangian function of (366) is

$$\mathcal{L}(\mathbf{r}, \lambda) = \|\bar{\boldsymbol{\epsilon}}_F(\mathbf{r})\|^2 - \lambda \cdot (\Delta^2 - \mathbf{r}^T \cdot \mathbf{r}) \qquad (367)$$

The stationary point of (367) is calculated as:

$$\nabla\mathcal{L}(\mathbf{r}, \lambda) \equiv \mathbf{0} : \qquad (\mathbf{J}^T \cdot \mathbf{J} + \lambda \cdot \mathbf{I}) \cdot \mathbf{r} = -\mathbf{J} \cdot \boldsymbol{\epsilon}_{F,0} \qquad (368)$$

The solution of (368) is known as Levenberg-Marquardt direction.

An actual least-squares/trust-region step $\mathbf{r}^*$ is determined by the following procedure:

1. Solve (368) starting from $\lambda = 0$ with increasing values of λ including the limit $\lambda \rightarrow \infty$.

2. Visualize the resulting Pareto front $PF_{\mathbf{r}^*}$ of all optimal objective value for all trust regions, $\|\bar{\boldsymbol{\epsilon}}_F(\mathbf{r}^*)\|^2$ vs. $\|\mathbf{r}^*\|^2$.

3. Select a step from the Pareto front $PF_{\mathbf{r}^*}$ utilizing its bend and additional simulations.

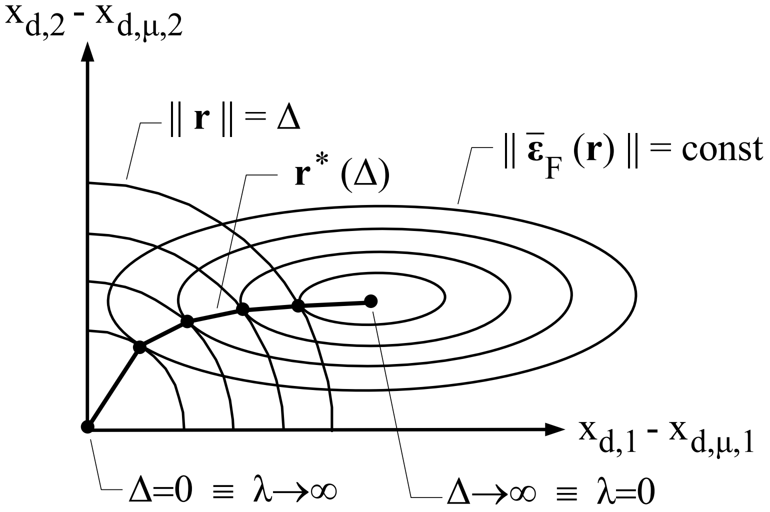

$x_{d,2} - x_{d,\mu,2}$
$\| \mathbf{r} \| = \Delta$
$\mathbf{r}^* (\Delta)$
$\| \bar{\varepsilon}_F (\mathbf{r}) \| = \text{const}$
$x_{d,1} - x_{d,\mu,1}$
$\Delta=0 \equiv \lambda\rightarrow\infty$
$\Delta\rightarrow\infty \equiv \lambda=0$

Figure 59. Least-squares/trust-region approach to geometric-yield optimization according to (366). Quarter-circles represent trust-regions of a step. Ellipses represent level contours of the least-squares optimization objective (363). $\mathbf{r}^*$ is the optimal step for the respective trust region determined by Δ.

The idea behind this procedure is to include the computation of a suitable trust region into the computation of a search direction. Figure 59 illustrates the procedure in the space of two design parameters $x_{d,1}$ and $x_{d,2}$. The two axes are shifted so that the origin lies in the actual design parameter vector $\mathbf{x}_{d,\mu}$. The quarter-circles represent trust regions for a length of step $\mathbf{r}$ not larger than Δ. The ellipses represent level contours of the least-squares optimization objective (363).

If the allowed step length Δ is zero, no step is allowed and the resulting step $\mathbf{r}^*$ has length zero. This corresponds to a Lagrange factor λ going to infinity in the Lagrangian function (368).

If the allowed step length is arbitrarily large, i.e. there is no restriction on the step length, this means that Δ is infinite and that λ is zero and (368) becomes (365). In this case, the minimum of $\|\bar{\epsilon}_F(\mathbf{r})\|^2$ will determine the step $\mathbf{r}^*$ as illustrated in Figure 59.

For an allowed step length in between, i.e. $0 < \Delta < \infty$, the resulting problem resembles the problem of a geometric yield analysis. The optimum step $\mathbf{r}^*$ results from the point where the bounding circle of the trust region touches

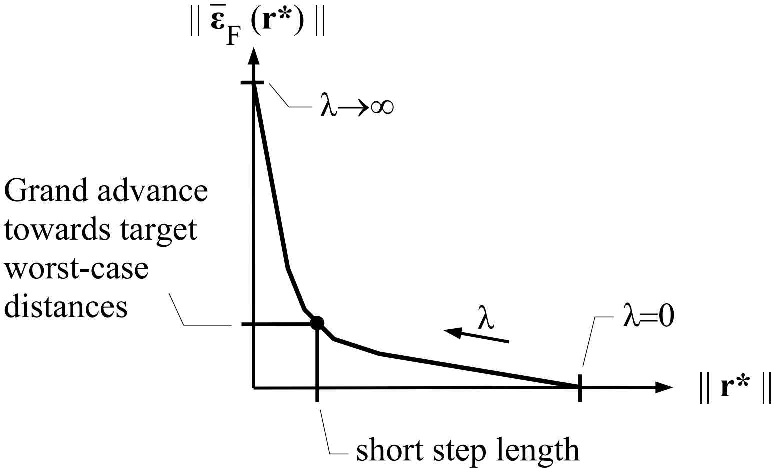


Figure 60. Pareto front of optimal solutions of least-squares/trust-region approach (366) in dependence of maximum steplength Δ, i.e. Lagrange factor λ (367). A point in the bend corresponds to a step with a grand progress towards the target worst-case distances at a small step length.

a level contour of the objective $\|\bar{\epsilon}_F(\mathbf{r})\|^2$. Figure 59 illustrates optimum steps for some trust regions.

Due to the quadratic nature of the objective $\|\bar{\epsilon}_F(\mathbf{r})\|^2$, the amount of additional decrease in the objective that can be obtained by an increase of the allowed step length $\|\mathbf{r}\| \leq \Delta$ is decreasing. This is even more so with a worsening problem condition of the Jacobian matrix $\mathbf{J}$.

Therefore, the Pareto front of objectives $\|\bar{\epsilon}_F(\mathbf{r}^*)\|^2$ versus the step $\mathbf{r}^*$ according to (366) acquires a typical shape as illustrated in Figure 60.

A point in the bend of this curve is preferable, because it leads to a grand progress towards the target worst-case distances at a small step length. The additional progress towards the target worst-case distances beyond the bend is rather small. Additionally, the required step length for additional progress beyond the bend becomes large, and the step will be more likely to leave the region of validity of the linearized model (362). Therefore, a step in the bend will be selected. Additional simulations have to be spent to verify that the linearized model holds in the corresponding trust region.

Note that the described approach is a part of a complete optimization algorithm. Other algorithmic components are required that deal with updating the target values of the optimization objectives, or with determining bends in the Pareto curve.

Efficient methods to calculate the Pareto curve of progress in the optimization objective versus step length according to problem (366) have been developed [7, 9]. They include the additional constraints in the problem formulation (359).

Special Case: Nominal Design. The least-squares/trust-region solution approach described in this section can be applied to nominal design as well.

The worst-case distances then are simply replaced by the performance features and corresponding target values.

7.2.4 Min-Max Solution Approach

According to the definition of yield and yield partitions, (137)–(139), the smallest yield partition value is an upper bound for the overall yield.

The multiple-objective geometric-yield optimization problem (359) can therefore be transformed into a single-objective optimization by maximization of the minimum yield partition:

$$\max_{\mathbf{x}_d} \min_i \alpha_i(\mathbf{x}_d) \cdot \beta_{WL/U,i}(\mathbf{x}_d) \quad \text{s.t.} \quad \mathbf{c}(\mathbf{x}_d) \geq \mathbf{0} \tag{369}$$

The optimization objective of problem formulation (369) is also obtained by using the l_∞-norm in (366).

(369) describes a maximum tolerance ellipsoid inside the parameter acceptance region as illustrated in Figure 61.

Sometimes, the min-max approach to geometric-yield optimization is denoted as design centering. Design centering in the sense of finding an equilibrium of the center of gravity of the truncated probability density function however is similar but not equal to geometric-yield optimization.

7.2.5 Linear-Programming Solution Approach

Using the linearization of the worst-case distances (362), the min-max formulation (369) can be transformed into a linear programming problem.

$$\max_{\mathbf{x}_d, \beta} \beta \ \text{ s.t. } \left\{ \begin{array}{l} \boldsymbol{\beta}_{WL/U}(\mathbf{x}_{d,\mu}) + \mathbf{J}(\mathbf{x}_{d,\mu}) \cdot (\mathbf{x}_d - \mathbf{x}_{d,\mu}) \geq \beta \cdot \mathbf{1} \\ \mathbf{c}(\mathbf{x}_{d,\mu}) + \nabla \mathbf{c}(\mathbf{x}_{d,\mu}^T) \cdot (\mathbf{x}_d - \mathbf{x}_{d,\mu}) \geq \mathbf{0} \end{array} \right. \tag{370}$$

$\mathbf{1}$ denotes a vector with “1” at each position. (370) formulates a worst-case distance value β that is to be maximized such that the linear models of all worst-case distances obtained from a geometric analysis have at least this value β. The resulting value of β will describe the largest tolerance ellipsoid that can be inscribed in the overall parameter acceptance region.

(370) describes a linear programming problem.

The geometric-yield optimization problem then consists in a sequence of linear programming problems (370), where the linearization of the worst-case distances is iteratively updated.

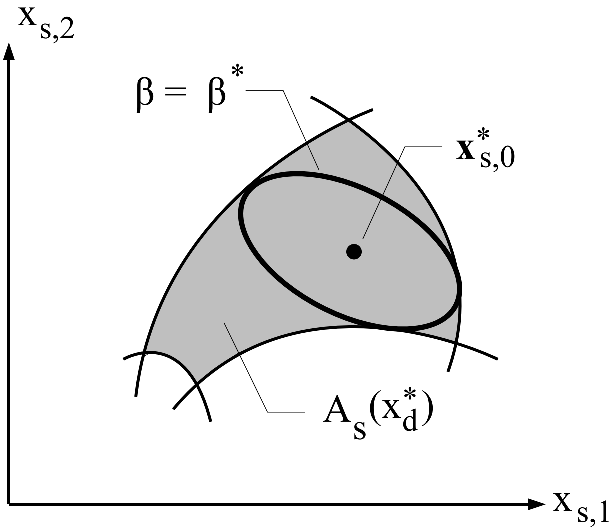


Figure 61. Min-max solution to geometric-yield optimization.

Note that any piecewise linear approximation of the parameter acceptance region can be applied within (370).

In [38], (370) has been presented for the mean values of statistical parameters as optimization parameters and coordinate searches to compute border points of the parameter acceptance region.

Ellipsoidal methods can be applied to solve (370) [1, 123].

7.2.6 Tolerance Assignment, Other Optimization Parameters

From (300) and (301), the first-order derivatives of the worst-case distance with regard to components of the covariance matrix **C**, i.e. standard deviations and correlations, which are the basis for tolerance assignment, can be derived [122].

In the same manner as described in Section 7.2.1, first-order derivatives of the worst-case distance with regard to any other quantity in problem (264) and (265), i.e. performance-feature bound, range-parameter bounds, can be derived [122].

Appendix A
Expectation Values

A.1 Expectation Value

The expectation value of a vector function[1] $\mathbf{h}(\mathbf{z})$ of a random vector[2] $\mathbf{z}$, $\mathbf{h} \in \mathbb{R}^{n_h}$, $\mathbf{z} \in \mathbb{R}^{n_z}$, that originates from a statistical distribution with the probability density function $\mathrm{pdf}(\mathbf{z})$ is denoted by and defined as:

$$\underset{\mathrm{pdf}(\mathbf{z})}{\mathrm{E}\left\{\mathbf{h}(\mathbf{z})\right\}} = \mathrm{E}\left\{\mathbf{h}(\mathbf{z})\right\} = \int_{-\infty}^{+\infty} \ldots \int_{-\infty}^{+\infty} \mathbf{h}(\mathbf{z}) \cdot \mathrm{pdf}(\mathbf{z}) \cdot d\mathbf{z} \tag{A.1}$$

$$d\mathbf{z} = dz_1 \cdot dz_2 \cdot \ldots \cdot dz_{n_z}$$

Given a probability density function $\mathrm{pdf}(\mathbf{z})$, the expectation value is assumed to refer to this probability density function without explicitly mentioning it.

A.2 Moments

The moment of order κ, $\mathrm{m}^{(\kappa)}$, of a single random variable z results from setting $\mathbf{h}(\mathbf{z}) = z^\kappa$ in (A.1):

$$\mathrm{m}^{(\kappa)} = \mathrm{E}\left\{z^\kappa\right\} \tag{A.2}$$

A.3 Mean Value

The moment of order 1, i.e. the expectation value of z, or $\mathbf{z}$ respectively, is denoted as mean value m, or $\mathbf{m}$ respectively:

$$\mathrm{m}^{(1)} = \mathrm{m} \quad = \quad \mathrm{E}\left\{z\right\} \tag{A.3}$$

[1] See Note 2 in Section 3.7.

[2] See Note 1 in Section 3.5.

$$\mathbf{m} \;=\; \mathrm{E}\{\mathbf{z}\} = [\mathrm{E}\{z_1\} \;\ldots\; \mathrm{E}\{z_{n_z}\}]^T \tag{A.4}$$

A.4 Central Moments

The central moment of order κ, $\mathrm{c}^{(\kappa)}$, of a single random variable z results from setting $\mathbf{h}(\mathbf{z}) = (z - \mathrm{m})^\kappa$ in (A.1):

$$\mathrm{c}^{(\kappa)} = \mathrm{E}\{(z - \mathrm{m})^\kappa\} \tag{A.5}$$

A.5 Variance

The variance of a single random variable z is defined as the central moment of order 2 of z and denoted by σ^2 or $\mathrm{V}\{z\}$:

$$\mathrm{c}^{(2)} = \sigma^2 = \mathrm{V}\{z\} = \mathrm{E}\left\{(z - \mu)^2\right\} \tag{A.6}$$

σ denotes the standard deviation of a single random variable z:

$$\sigma = \sqrt{\mathrm{V}\{z\}} \tag{A.7}$$

A.6 Covariance

The covariance, $\mathrm{cov}\{z_k, z_l\}$, of two random variables z_k and z_l is defined as a mixed central moment of order 2:

$$\mathrm{cov}\{z_k, z_l\} = \mathrm{E}\{(z_k - \mathrm{m}_k) \cdot (z_l - \mathrm{m}_l)\} \tag{A.8}$$

A.7 Correlation

The correlation, $\mathrm{corr}\{z_k, z_l\}$, of two random variables z_k and z_l is defined as their covariance normalized with respect to their individual standard deviations:

$$\mathrm{corr}\{z_k, z_l\} = \frac{\mathrm{cov}\{z_k, z_l\}}{\sigma_k \cdot \sigma_l} \tag{A.9}$$

A.8 Variance/Covariance Matrix

The central moments of order 2 of a vector $\mathbf{z}$ are defined component-wise according to (A.6) and (A.8) and combined in the variance/covariance matrix or simply covariance matrix $\mathbf{C} = \mathrm{V}\{\mathbf{z}\}$:

$$\begin{aligned}\mathbf{C} &= \mathrm{V}\{\mathbf{z}\} = \mathrm{E}\left\{(\mathbf{z} - m) \cdot (\mathbf{z} - m)^T\right\} \\ &= \begin{bmatrix} \mathrm{V}\{z_1\} & \mathrm{cov}\{z_1, z_2\} & \ldots & \mathrm{cov}\{z_1, z_{n_z}\} \\ \mathrm{cov}\{z_2, z_1\} & \mathrm{V}\{z_2\} & \ldots & \mathrm{cov}\{z_2, z_{n_z}\} \\ \vdots & & & \vdots \\ \mathrm{cov}\{z_{n_z}, z_1\} & \ldots & & \mathrm{V}\{z_{n_z}\} \end{bmatrix}\end{aligned} \tag{A.10}$$

Correspondingly, the covariance matrix of a vector function $\mathbf{h}(\mathbf{z})$ of a random vector $\mathbf{z}$ is defined as:

$$\mathrm{V}\left\{\mathbf{h}(\mathbf{z})\right\} = \mathrm{E}\left\{(\mathbf{h}(\mathbf{z}) - \mathrm{E}\left\{\mathbf{h}(\mathbf{z})\right\}) \cdot (\mathbf{h}(\mathbf{z}) - \mathrm{E}\left\{\mathbf{h}(\mathbf{z})\right\})^T\right\} \tag{A.11}$$

A.9 Calculation Formulas

In the following, 3 calculation formulas are given together with respective special cases.

$$\begin{aligned}
\mathrm{E}\left\{\mathbf{A}\cdot\mathbf{h}(\mathbf{z})+\mathbf{b}\right\} &= \mathbf{A}\cdot\mathrm{E}\left\{\mathbf{h}(\mathbf{z})\right\}+\mathbf{b} \qquad \text{(A.12)}\\
\mathrm{E}\left\{\mathrm{const}\right\} &= \mathrm{const}\\
\mathrm{E}\left\{c\cdot\mathbf{h}(\mathbf{z})\right\} &= c\cdot\mathrm{E}\left\{\mathbf{h}(\mathbf{z})\right\}\\
\mathrm{E}\left\{h_1(\mathbf{z})+h_2(\mathbf{z})\right\} &= \mathrm{E}\left\{h_1(\mathbf{z})\right\}+\mathrm{E}\left\{h_2(\mathbf{z})\right\}
\end{aligned}$$

$$\begin{aligned}
\mathrm{V}\left\{\mathbf{A}\cdot\mathbf{h}(\mathbf{z})+\mathbf{b}\right\} &= \mathbf{A}\cdot\mathrm{V}\left\{\mathbf{h}(\mathbf{z})\right\}\cdot\mathbf{A}^T \qquad \text{(A.13)}\\
\mathrm{V}\left\{\mathbf{a}^T\cdot\mathbf{h}(\mathbf{z})+\mathbf{b}\right\} &= \mathbf{a}^T\cdot\mathrm{V}\left\{h(\mathbf{z})\right\}\cdot\mathbf{a}\\
\mathrm{V}\left\{a\cdot h(\mathbf{z})+b\right\} &= a^2\cdot\mathrm{V}\left\{h(\mathbf{z})\right\}
\end{aligned}$$

$$\begin{aligned}
\mathrm{V}\left\{\mathbf{h}(\mathbf{z})\right\} &= \mathrm{E}\left\{(\mathbf{h}(\mathbf{z})-\mathbf{a})\cdot(\mathbf{h}(\mathbf{z})-\mathbf{a})^T\right\}\\
&\quad -(\mathrm{E}\left\{\mathbf{h}(\mathbf{z})\right\}-\mathbf{a})\cdot(\mathrm{E}\left\{\mathbf{h}(\mathbf{z})\right\}-\mathbf{a})^T \qquad \text{(A.14)}\\
\mathrm{V}\left\{h(\mathbf{z})\right\} &= \mathrm{E}\left\{(h(\mathbf{z})-a)^2\right\}-(\mathrm{E}\left\{h(\mathbf{z})\right\}-a)^2\\
\mathrm{V}\left\{\mathbf{h}(\mathbf{z})\right\} &= \mathrm{E}\left\{\mathbf{h}(\mathbf{z})\cdot\mathbf{h}^T(\mathbf{z})\right\}-\mathrm{E}\left\{\mathbf{h}(\mathbf{z})\right\}\cdot\mathrm{E}\left\{\mathbf{h}^T(\mathbf{z})\right\}\\
\mathrm{V}\left\{h(\mathbf{z})\right\} &= \mathrm{E}\left\{h^2(\mathbf{z})\right\}-(\mathrm{E}\left\{h(\mathbf{z})\right\})^2
\end{aligned}$$

(A.12) is the linear transformation formula for expectation values. It says that the expectation value of the linear transformation of a random vector equals the corresponding linear transformation of the vector's expectation value.

(A.13) is the linear transformation formula for variances.

(A.14) is the translation formula for variances. It relates the variance as a second-order central moment to the second-order moment and the quadratic expectation value.

From (A.13) follows the Gaussian error propagation:

$$\begin{aligned}
\mathrm{V}\{\mathbf{a}^T\cdot\mathbf{z}+\mathbf{b}\} &= \mathbf{a}^T\cdot\mathrm{V}\left\{\mathbf{z}\right\}\cdot\mathbf{a} = \mathbf{a}^T\cdot\mathbf{C}\cdot\mathbf{a}\\
&= \sum_{k,l} a_k a_l \sigma_k \varrho_{k,l} \sigma_l \overset{\varrho_{k,l}=0}{=} \sum_k a_k^2 \sigma_k^2 \qquad \text{(A.15)}
\end{aligned}$$

A.10 Standardization of Random Variables

A random variable z originating from a statistical distribution corresponds to a standardized random variable z' with a mean value of zero and a variance of one.

The standardization is done with the following formula:

$$\begin{aligned} z' &= \frac{z - \mathrm{E}\{z\}}{\sqrt{\mathrm{V}\{z\}}} = \frac{z - m_z}{\sigma_z} \qquad \text{(A.16)} \\ \mathrm{E}\{z'\} &= 0 \\ \mathrm{V}\{z'\} &= 1 \end{aligned}$$

A.11 Exercises

1. Prove (A.12). (Apply (A.1), (15).)

2. Prove (A.13). (Apply (A.11), (A.12), (A.1).)

3. Prove (A.14). (Apply (A.11), (A.12).)

4. Prove that z' according to (A.16) has a mean value of zero and a variance of one. (Apply calculation formulas from Section A.9.)

5. Show that (52) holds for a multinormal distribution according to (23) and (24). (Insert (23) and (24) in (A.1). Use (57) and (58) for a variable substitution. Consider that the corresponding integral over an even function is zero.)

6. Show that (53) holds for a multinormal distribution according to (23) and (24). (Insert (23) and (24) in (A.10). Use (57) and (58) for a variable substitution.)

Appendix B
Statistical Estimation of Expectation Values

B.1 Expectation-Value Estimator

A consistent, unbiased of the expectation value (A.1) is

$$\widehat{\mathrm{E}}\left\{\mathbf{h}(\mathbf{x})\right\} = \widehat{\mathbf{m}}_h = \frac{1}{n_{MC}} \sum_{\mu=1}^{n_{MC}} \mathbf{h}(\mathbf{x}^{(\mu)}) \tag{B.1}$$

$$\mathbf{x}^{(\mu)} \sim \mathcal{D}\left(\mathrm{pdf}(\mathbf{x})\right),\ \mu = 1, \ldots, n_{MC}$$

A statistical estimation is based on a sample of sample elements.

A widehat is used to denote an estimator function $\widehat{\mathbf{\Phi}}(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n_{MC})})$ for a function $\mathbf{\Phi}(\mathbf{x})$.

Sample elements $\mathbf{x}^{(\mu)}$ are independently and identically distributed according to the given statistical distribution $\mathcal{D}$ with the probability density function $\mathrm{pdf}(\mathbf{x})$. Therefore:

$$\mathrm{E}\left\{\mathbf{h}(\mathbf{x}^{(\mu)})\right\} = \mathrm{E}\left\{\mathbf{h}(\mathbf{x}\right\} = \mathbf{m}_h \tag{B.2}$$

$$\mathrm{V}\left\{[\mathbf{h}(\mathbf{x}^{(1)}) \ldots \mathbf{h}(\mathbf{x}^{(n_{MC})})]^T\right\} = \mathrm{diag}\left(\mathrm{V}\{\mathbf{h}(\mathbf{x}^{(1)})\} \ldots \mathrm{V}\{\mathbf{h}(\mathbf{x}^{(n_{MC})})\}\right) \tag{B.3}$$

$$\mathrm{V}\left\{\mathbf{h}(\mathbf{x}^{(\mu)})\right\} = \mathrm{V}\left\{\mathbf{h}(\mathbf{x})\right\} \tag{B.4}$$

B.2 Variance Estimator

A consistent, unbiased estimator of the variance/covariance matrix (A.11) is

$$\widehat{\mathrm{V}}\left\{\mathbf{h}(\mathbf{x})\right\} = \frac{1}{n_{MC}-1} \sum_{\mu=1}^{n_{MC}} \left(\mathbf{h}(\mathbf{x}^{(\mu)}) - \widehat{\mathbf{m}}_h\right) \cdot \left(\mathbf{h}(\mathbf{x}^{(\mu)}) - \widehat{\mathbf{m}}_h\right)^T \tag{B.5}$$

$$\mathbf{x}^{(\mu)} \sim \mathcal{D}\left(\mathrm{pdf}(\mathbf{x})\right),\ \mu = 1, \ldots, n_{MC}$$

In (B.5), the estimated expectation value $\widehat{\mathbf{m}}_h$ is used. If the expectation value $\mathbf{m}_h$ would be used, the consistent, unbiased estimator would be:

$$\widehat{\widehat{\mathrm{V}}}\left\{\mathbf{h}(\mathbf{x})\right\} = \frac{1}{n_{MC}} \sum_{\mu=1}^{n_{MC}} \left(\mathbf{h}(\mathbf{x}^{(\mu)}) - \mathbf{m}_h\right) \cdot \left(\mathbf{h}(\mathbf{x}^{(\mu)}) - \mathbf{m}_h\right)^T \tag{B.6}$$

B.3 Estimator Bias

The bias $\mathbf{b}$ of an estimator is the difference between the expectation value of the estimator function and the original function:

$$\mathbf{b}_{\widehat{\Phi}} = \mathrm{E}\left\{\widehat{\mathbf{\Phi}}(\mathbf{x})\right\} - \mathbf{\Phi}(\mathbf{x}) \tag{B.7}$$

The expectation value of an unbiased estimator function is the original function:

$$\widehat{\mathbf{\Phi}}(\mathbf{x}) \text{ is unbiased } \Leftrightarrow \mathbf{b}_{\widehat{\Phi}} = \mathbf{0} \Leftrightarrow \mathrm{E}\left\{\widehat{\mathbf{\Phi}}(\mathbf{x})\right\} = \mathbf{\Phi}(\mathbf{x}) \tag{B.8}$$

A consistent estimator function certainly approaches the original function if the number of sample elements n_{MC} approaches ∞:

$$\widehat{\mathbf{\Phi}}(\mathbf{x}) \text{ is consistent } \Leftrightarrow \text{prob}\left\{\lim_{n_{MC}\to\infty} \widehat{\mathbf{\Phi}} = \mathbf{\Phi}\right\} = 1 \tag{B.9}$$

B.4 Estimator Variance

A measure for the quality of an estimator function is the 2nd-order moment of the estimator function around the original function:

$$\mathbf{Q}_{\widehat{\Phi}} = \mathrm{E}\left\{(\widehat{\mathbf{\Phi}} - \mathbf{\Phi}) \cdot (\widehat{\mathbf{\Phi}} - \mathbf{\Phi})^T\right\} = \mathrm{V}\left\{\widehat{\mathbf{\Phi}}\right\} + \mathbf{b}_{\widehat{\Phi}} \cdot \mathbf{b}_{\widehat{\Phi}}^T \tag{B.10}$$

From (B.10) follows that the quality measure $\mathbf{Q}$ is the estimator variance, i.e. the variance of the estimation function, for an unbiased estimator:

$$\mathbf{b}_{\widehat{\Phi}} = \mathbf{0} : \quad \mathbf{Q}_{\widehat{\Phi}} = \mathrm{V}\left\{\widehat{\mathbf{\Phi}}\right\} \tag{B.11}$$

B.5 Expectation-Value-Estimator Variance

To determine the quality of the expectation-value estimator of (B.1), (B.11) can be applied, because (B.1) provides an unbiased estimator (see Exercise B.7.1).

$$\mathbf{Q}_{\widehat{\mu}_h} = \mathrm{V}\left\{\widehat{\mathrm{E}}\left\{\mathbf{h}(\mathbf{x})\right\}\right\} \tag{B.12}$$

Applying (B.1) and (A.13) leads to

$$\mathbf{Q}_{\widehat{\mu}_h} = \frac{1}{n_{MC}} \cdot \mathrm{V}\left\{ \left[1\ 1\ \ldots\ 1\right] \cdot \left[\mathbf{h}^{(1)}\ \mathbf{h}^{(2)}\ \ldots\ \mathbf{h}^{(n_{MC})}\right]^T\right\}, \tag{B.13}$$

with $\mathbf{h}^{(\mu)} = \mathbf{h}(\mathbf{x}^{(\mu)})$. Applying (A.13), (B.3) and (B.4) finally leads to:

$$\mathbf{Q}_{\widehat{\mu}_h} = \mathrm{V}\left\{\widehat{\mathrm{E}}\left\{\mathbf{h}(\mathbf{x})\right\}\right\} = \frac{1}{n_{MC}} \cdot \mathrm{V}\left\{\mathbf{h}(\mathbf{x})\right\} \tag{B.14}$$

Using (B.16) and (B.17) in the above proof leads to the corresponding formula for an estimator of the quality:

$$\widehat{\mathbf{Q}}_{\widehat{\mu}_h} = \widehat{\mathrm{V}}\left\{\widehat{\mathrm{E}}\left\{\mathbf{h}(\mathbf{x})\right\}\right\} = \frac{1}{n_{MC}} \cdot \widehat{\mathrm{V}}\left\{\mathbf{h}(\mathbf{x})\right\} \tag{B.15}$$

B.6 Estimator Calculation Formulas

In the following, estimator formulas corresponding to the calculation formulas (A.12), (A.13) and (A.14) are given.

$$\widehat{\mathrm{E}}\left\{\mathbf{A} \cdot \mathbf{h}(\mathbf{z}) + \mathbf{b}\right\} = \mathbf{A} \cdot \widehat{\mathrm{E}}\left\{\mathbf{h}(\mathbf{z})\right\} + \mathbf{b} \tag{B.16}$$

$$\widehat{\mathrm{V}}\left\{\mathbf{A} \cdot \mathbf{h}(\mathbf{z}) + \mathbf{b}\right\} = \mathbf{A} \cdot \widehat{\mathrm{V}}\left\{\mathbf{h}(\mathbf{z})\right\} \cdot \mathbf{A}^T \tag{B.17}$$

$$\widehat{\mathrm{V}}\left\{\mathbf{h}(\mathbf{z})\right\} = \frac{n_{MC}}{n_{MC} - 1}\left(\widehat{\mathrm{E}}\left\{\mathbf{h}(\mathbf{z}) \cdot \mathbf{h}^T(\mathbf{z})\right\} - \widehat{\mathrm{E}}\left\{\mathbf{h}(\mathbf{z})\right\} \cdot \widehat{\mathrm{E}}\left\{\mathbf{h}^T(\mathbf{z})\right\}\right) \tag{B.18}$$

B.7 Exercises

1. Prove that the expectation-value estimator according to (B.1) is unbiased. (Check if (B.8) is true. Apply (A.12) and (B.2).)

2. Prove the second part of (B.10). (Express $\mathbf{\Phi}$ by using (B.7) and insert in (A.14). Combine terms with $\widehat{\mathbf{\Phi}}$ as h.)

3. Prove that the variance estimator according to (B.5) is unbiased.
Check if (B.8) is true. Apply (B.5). Within the sum extend each product term by $-\mathbf{m}_h + \mathbf{m}_h$. Apply (B.1) to get to a form with a sum of the dyadic product of $\mathbf{h}^{(\mu)} - \mathbf{m}_h$ and n_{MC} times the dyadic product of $\widehat{\mathbf{m}}_h - \mathbf{m}_h$. Then apply (B.4) and (B.14).

4. Prove (B.16).

5. Prove (B.17).

6. Prove (B.18).

Appendix C
Optimality Conditions of Nonlinear Optimization Problems

Optimality conditions of optimization problems can be found in the literature [56, 45, 91, 19]. The following summary adopts the descriptions in [45].

The statements in the following hold for a local optimum. Global optimization requires a preceding stage to identify regions of local optima.

C.1 Unconstrained Optimization

Without loss of generality, we will assume that a minimum of a function $f(\mathbf{x})$ shall be targeted in the following:

$$\min_{\mathbf{x}} f(\mathbf{x}) \tag{C.1}$$

Starting point is the Taylor expansion of the objective function $f(\mathbf{x})$ at the optimum solution $\mathbf{x}^*$ and $f^* = f(\mathbf{x}^*)$:

$$\begin{aligned} f(\mathbf{x}) &= \underbrace{f(\mathbf{x}^*)}_{f^*} + \underbrace{\nabla f(\mathbf{x}^*)^T}_{\mathbf{g}^T} \cdot (\mathbf{x} - \mathbf{x}^*) \\ &\quad + \frac{1}{2} \cdot (\mathbf{x} - \mathbf{x}^*)^T \cdot \underbrace{\nabla^2 f(\mathbf{x}^*)}_{\mathbf{H}} \cdot (\mathbf{x} - \mathbf{x}^*) + \ldots \end{aligned} \tag{C.2}$$

With $\mathbf{x} = \mathbf{x}^* + \mathbf{r}$, a point $\mathbf{x}$ in the direction $\mathbf{r}$ from the optimum solution $\mathbf{x}^*$ has the objective value:

$$f(\mathbf{x}^* + \mathbf{r}) = f^* + \mathbf{g}^T \cdot \mathbf{r} + \frac{1}{2} \cdot \mathbf{r}^T \cdot \mathbf{H} \cdot \mathbf{r} + \ldots \tag{C.3}$$

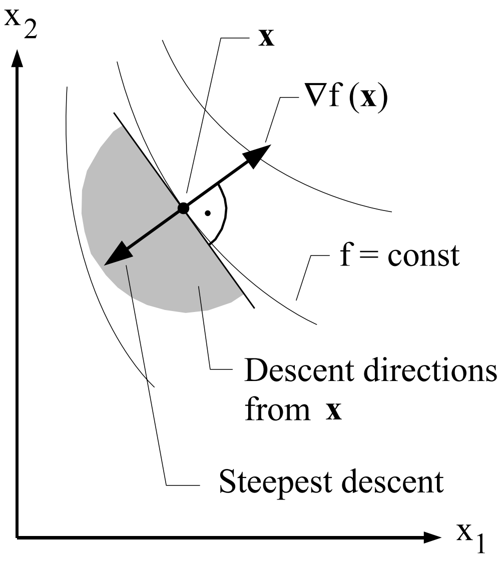


Figure C1. Descent directions and steepest descent direction of a function of two parameters.

C.2 First-Order Unconstrained Optimality Condition

Descent Direction. A descent direction $\mathbf{r}$ at any point $\mathbf{x}$ is defined by:

$$\Delta f(\mathbf{r}) = \nabla f(\mathbf{x})^T \cdot \mathbf{r} < 0 \tag{C.4}$$

Figure C1 illustrates the range of possible descent directions for an example with two parameters. All vectors starting from $\mathbf{x}$ that are in the gray region are descent directions. The steepest descent direction is the direction in which the amount of decrease in the objective function is maximum. From (C.4) follows that the steepest descent is along the negative gradient (Figure C1):

$$\min_{\mathbf{r}} \Delta f(\mathbf{r}) = \nabla f(\mathbf{x})^T \cdot (-\nabla f(\mathbf{x})) \tag{C.5}$$

Minimum: No More Descent. $\mathbf{x}^*$ being the solution with minimum objective value f^* requires that there exists no descent direction $\mathbf{r}$ starting from $\mathbf{x}^*$. Only then, no further decrease in f is achievable and f^* is the minimum:

$$\underset{\mathbf{r} \neq \mathbf{0}}{\forall} \quad \mathbf{g}^T \cdot \mathbf{r} \geq 0 \tag{C.6}$$

Necessary Condition. The only way to satisfy (C.6) is that the gradient $\mathbf{g}$ is zero. That represents the necessary first-order optimality condition for a local minimum solution $\mathbf{x}^*$ with $f^* = f(\mathbf{x}^*)$:

$$\nabla f(\mathbf{x}^*) = \mathbf{0} \tag{C.7}$$

A point that satisfies (C.7) is also called stationary point.

C.3 Second-Order Unconstrained Optimality Condition

Inserting (C.7) into (C.3), we can:

see that another condition for the quadratic term has to be formulated, in order to guarantee that no decrease in f is achievable at $\mathbf{x}^*$. This condition says that no direction $\mathbf{r}$ may exist, in which the quadratic term in (C.3) leads to a reduction in the objective value $f^* = f(\mathbf{x}^*)$.

Necessary Condition. The requirement of no further reduction is a necessary second-order optimality condition for a local minimum solution $\mathbf{x}^*$ with $f^* = f(\mathbf{x}^*)$.

$$\underset{\mathbf{r} \neq \mathbf{0}}{\forall} \quad \mathbf{r}^T \cdot \nabla^2 f(\mathbf{x}^*) \cdot \mathbf{r} \geq 0 \quad \Leftrightarrow \quad \nabla^2 f(\mathbf{x}^*) \text{ is positive semidefinite} \tag{C.8}$$

Figure C2 illustrates examples of functions of two parameters featuring different types of definiteness of their second-order derivatives.

We can see that positive definiteness refers to a minimum, negative definiteness to a maximum. If the Hessian is indefinite, no finite optimum exists. If the Hessian is positive semidefinite, no more descent can happen, but there may be no unique solution.

The type of definiteness of the Hessian corresponds to the signs of its eigenvalues. Positive definiteness corresponds to positive eigenvalues, negative definiteness to negative eigenvalues. Positive semidefiniteness corresponds to eigenvalues greater or equal zero. Indefiniteness corresponds to eigenvalues both negative and positive.

Sufficient Condition. A sufficient second-order optimality condition for a local minimum solution $\mathbf{x}^*$ with $f^* = f(\mathbf{x}^*)$ is that only an increase in the objective around the minimum is possible:

$$\underset{\mathbf{r} \neq \mathbf{0}}{\forall} \quad \mathbf{r}^T \cdot \nabla^2 f(\mathbf{x}^*) \cdot \mathbf{r} > 0 \quad \Leftrightarrow \quad \nabla^2 f(\mathbf{x}^*) \text{ is positive definite} \tag{C.9}$$

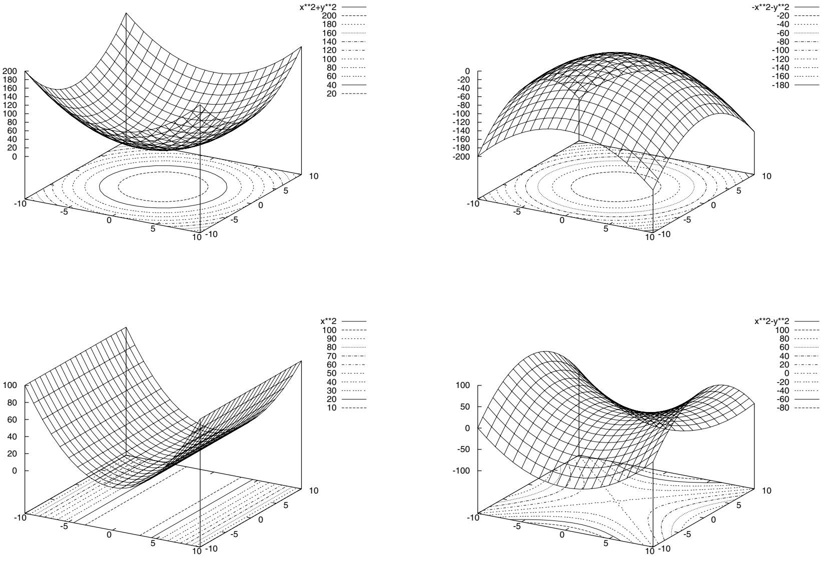


Figure C2. Different types of definiteness of the second-derivative of a function of two parameters: positive definite (upper left), positive semidefinite (upper right), indefinite (lower left), negative definite (lower right).

C.4 Constrained Optimization

Without loss of generality, we will assume that a minimum of a function $f(\mathbf{x})$ shall be targeted under equality and inequality constraints in the following form

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to} \quad \left\{ \begin{array}{ll} c_\mu(\mathbf{x}) = 0, & \mu \in EC \\ c_\mu(\mathbf{x}) \geq 0, & \mu \in IC \end{array} \right. \tag{C.10}$$

EC and IC denote the index sets of equality and inequality constraints.

The optimality conditions will be based on the Lagrangian function of (C.10) that combines objective and constraints into one new objective:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) \; - \sum_{\mu \in EC \cup IC} \lambda_\mu \cdot c_\mu(\mathbf{x}) \tag{C.11}$$

λ_μ is called Lagrange factor. It is part of the optimization parameters and has a certain optimum value.

Note that in (C.10) we have chosen to formulate "greater-than" constraints and to subtract the constraint components in the Lagrangian function (C.11). This determines the sign of the optimum Lagrange factor.

As in the unconstrained case, conditions will be formulated for the first- and second-order derivative that ensure that no further descent can happen in the surrounding of a minimum. But now, they are extended by the consideration of the constraints in the Lagrangian function.

C.5 First-Order Constrained Optimality Condition

Unconstrained Descent Direction. In constrained optimization, the concept of descent direction is extended to unconstrained descent directions. An unconstrained descent direction $\mathbf{r}$ with respect to one constraint c_μ at any point $\mathbf{x}$ is defined by:

$$\Delta f(\mathbf{r}) = \nabla f(\mathbf{x})^T \cdot \mathbf{r} < 0 \tag{C.12}$$

$$c_\mu(\mathbf{x} + \mathbf{r}) \approx c_\mu(\mathbf{x}) + \nabla c_\mu(\mathbf{x})^T \cdot \mathbf{r} \geq 0 \tag{C.13}$$

(C.12) is identical to the unconstrained case (C.4). It describes the directions $\mathbf{r}$ in which the objective will decrease.

(C.13) describes the restrictions due to the constraint c_μ. If the constraint is not active at point $\mathbf{x}$, then there is no restriction on the direction $\mathbf{r}$, only on the length of it. The constraint being inactive means that it is still away from the bound. Then, a descent direction may eventually result in the constraint value reaching the bound. In order not to violate the constraint, the descent direction may not exceed a certain length ϵ:

$$c_\mu(\mathbf{x}) > 0 : \qquad \|\mathbf{r}\| \leq \epsilon \tag{C.14}$$

If the constraint is active at point $\mathbf{x}$, then the value of c_μ is at its limit, $c_\mu = 0$. Equality constraints hence are permanently active, inequality constraints may become active or inactive again. Then, there is a restriction on the direction $\mathbf{r}$. Only those directions are allowed according to (C.13) in which the constraint value stays at its limit or in which the constraint value is increased such that the constraint will become inactive again.

Figure C3 illustrates the range of possible descent directions for an example with two parameters with a gray block sector of larger radius.

The performance contour $f = \text{const}$ is the same as in Figure C1. The performance gradient $\nabla f(\mathbf{x})$ defines a range of directions that are descent directions. In addition, a constraint is active, $c(\mathbf{x}) = 0$. The gradient of the constraint function, $\nabla c(\mathbf{x})$ defines a range of directions that are allowed. This range is illustrated with a gray block sector of smaller radius in Figure C3.

The intersection of the region of descent directions and the region of unconstrained directions is the region of unconstrained descent directions. In Figure

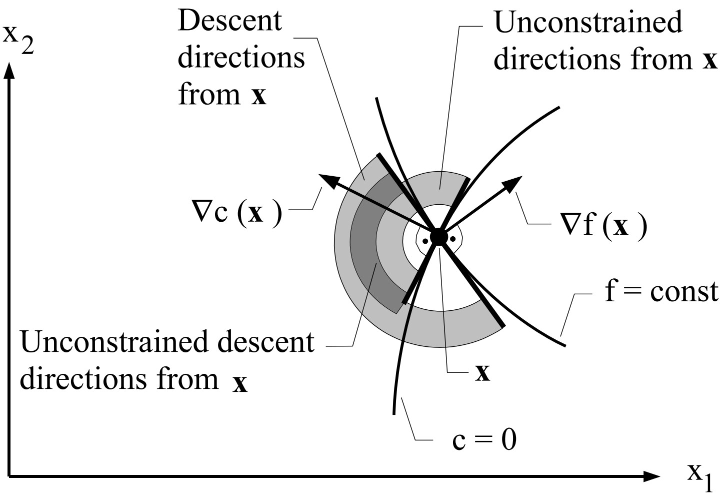

x2
Descent directions from **x**
Unconstrained directions from **x**
∇c (**x**)
∇f (**x**)
f = const
Unconstrained descent directions from **x**
x
c = 0
x1

Figure C3. Descent directions and unconstrained directions of a function of two parameters with one active constraint.

C3, this corresponds to the region where the two gray block sectors overlap and is illustrated with a dark gray block sector.

Obviously, each active constraint reduces the range of unconstrained descent directions.

Minimum: No More Unconstrained Descent. If $\mathbf{x}^*$ is the solution that achieves the minimum objective value f^* without violating any constraint, then there must not be any unconstrained descent direction $\mathbf{r}$ starting from $\mathbf{x}^*$. Only then, no further decrease in f is achievable and f^* is the minimum.

From Figure C3, we can motivate that no unconstrained descent region exists, if the dark gray region is empty. This is the case, if the gradients of objective and constraint are parallel and have the same direction:

$$\nabla f(\mathbf{x}^*) = \lambda^* \cdot \nabla c(\mathbf{x}^*) \quad \wedge \quad \lambda^* > 0 \tag{C.15}$$

If λ^* would have a negative sign, the gradients of objective and constraint would point in opposite directions. According to Figure C3, then the constraint would impose no restriction at all on the descent direction. Note that the positive sign is due to fact that we have formulated a minimization problem with lower bounds as constraints.

The first part of (C.15) could be derived by starting from the Lagrangian function (C.11) with only one constraint and treating it as done in the unconstrained case. Then, we would formulate the requirement that the first-order derivative of the gradient of the Lagrangian function is zero:

$$\nabla\mathcal{L}(\mathbf{x}) \equiv \mathbf{0} : \qquad \nabla f(\mathbf{x}^*) - \lambda^* \cdot \nabla c(\mathbf{x}^*) = \mathbf{0} \tag{C.16}$$

Necessary Condition. We can now formulate the necessary first-order optimality condition of: a constrained optimization problem according to (C.10) by formulating the no-descent condition through (C.15) and (C.16), and by adding the requirements on the constraint satisfaction:

$$\nabla\mathcal{L}(\mathbf{x}^*) = \nabla f(\mathbf{x}^*) \; - \sum_{\mu\in\mathcal{A}(\mathbf{x}^*)} \lambda_\mu^* \cdot \nabla c_\mu(\mathbf{x}^*) \; = \; \mathbf{0} \tag{C.17}$$

$$\mathcal{A}(\mathbf{x}^*) = EC \cup \{\mu \in IC \mid c_\mu(\mathbf{x}^*) = 0\}$$

$$c_\mu(\mathbf{x}^*) \; = \; 0 \, , \;\; \mu \in EC \tag{C.18}$$

$$c_\mu(\mathbf{x}^*) \; \geq \; 0 \, , \;\; \mu \in IC \tag{C.19}$$

$$\lambda_\mu^* \; \geq \; 0 \, , \;\; \mu \in IC \tag{C.20}$$

$$\lambda_\mu^* \cdot c_\mu(\mathbf{x}^*) \; = \; 0 \, , \;\; \mu \in EC \cup IC \tag{C.21}$$

(C.17) and (C.20) are explained through the condition that no unconstrained descent direction exists in the minimum. The restrictions in (C.17) are defined for all active constraints $\mathcal{A}(\mathbf{x}^*)$.

No statement about the sign of the Lagrange factor can be made for an equality constraint.

(C.18) and (C.19) formulate that the constraint must not be violated in the minimum.

(C.21) is the so-called complementarity condition. It expresses that in the minimum either a constraint is 0 (that means "active") or the corresponding Lagrange factor is 0. If both are zero at the same time, this corresponds to a minimum of the objective function where the constraint just got active. Deleting this constraint would not change the solution of the optimization problem.

A Lagrange factor of zero for an inactive constraint corresponds to deleting it from the Lagrange function. Therefore we have that:

$$\mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) \;\; = \;\; f(\mathbf{x}^*) \tag{C.22}$$

(C.17)–(C.21) are known as Karush-Kuhn-Tucker(KKT) conditions.

C.6 Second-Order Constrained Optimality Condition

The second-order optimality condition for the minimum f^* of a constrained optimization problem can be explained based on the quadratic form of the objective around the optimum $\mathbf{x}^*$ in a direction $\mathbf{r}$:

$$f(\mathbf{x}^* + \mathbf{r}) \overset{(C.22)}{=} \mathcal{L}(\mathbf{x}^* + \mathbf{r}, \boldsymbol{\lambda}^*)$$

$$= \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*) + \underbrace{\nabla\mathcal{L}(\mathbf{x}^*)}_{\mathbf{0}\ (C.17)}{}^T \cdot \mathbf{r} + \frac{1}{2}\mathbf{r}^T \cdot \nabla^2\mathcal{L}(\mathbf{x}^*) \cdot \mathbf{r} + \ldots \tag{C.23}$$

$$\nabla^2\mathcal{L}(\mathbf{x}^*) = \nabla^2 f(\mathbf{x}^*) - \sum_{\mu \in \mathcal{A}(\mathbf{x}^*)} \lambda_\mu^* \cdot \nabla^2 c_\mu(\mathbf{x}^*) \tag{C.24}$$

As in the unconstrained case, we require that no direction **r** may exist in which the quadratic term in (C.23) leads to a reduction in the objective value $f^* = f(\mathbf{x}^*)$. The difference to the unconstrained case is that we only have to consider unconstrained directions, but that the curvatures of both the objective function and the constraint functions are considered.

Necessary Condition. For any unconstrained stationary direction **r** from the stationary point $\mathbf{x}^*$, no descent is achievable through the second-order derivative of the Lagrangian function:

$$\forall_{\substack{\nabla c_\mu(\mathbf{x}^*)^T \cdot \mathbf{r} = 0\ ,\ \ \mu \in \mathcal{A}_+(\mathbf{x}^*) \\ \nabla c_\mu(\mathbf{x}^*)^T \cdot \mathbf{r} \geq 0\ ,\ \ \mu \in \mathcal{A}(\mathbf{x}^*) \setminus \mathcal{A}_+(\mathbf{x}^*) \\ \mathcal{A}_+(\mathbf{x}^*) = EC \cup \{\mu \in IC \,|\, c_\mu(\mathbf{x}^*) = 0 \wedge \lambda_\mu^* > 0\}}} \mathbf{r}^T \cdot \nabla^2\mathcal{L}(\mathbf{x}^*) \cdot \mathbf{r} \geq 0 \tag{C.25}$$

Note that this is a weaker requirement than positive semidefiniteness, because not all directions **r** are included in (C.25).

Sufficient Condition. The corresponding sufficient condition is that only an increase in the objective function is obtained into unconstrained stationary directions around the minimum:

$$\forall_{\substack{\nabla c_\mu(\mathbf{x}^*)^T \cdot \mathbf{r} = 0\ ,\ \ \mu \in \mathcal{A}_+(\mathbf{x}^*) \\ \nabla c_\mu(\mathbf{x}^*)^T \cdot \mathbf{r} \geq 0\ ,\ \ \mu \in \mathcal{A}(\mathbf{x}^*) \setminus \mathcal{A}_+(\mathbf{x}^*) \\ \mathcal{A}_+(\mathbf{x}^*) = EC \cup \{\mu \in IC \,|\, c_\mu(\mathbf{x}^*) = 0 \wedge \lambda_\mu^* > 0\}}} \mathbf{r}^T \cdot \nabla^2\mathcal{L}(\mathbf{x}^*) \cdot \mathbf{r} > 0 \tag{C.26}$$

Note that this is a weaker requirement than positive definiteness, because not all directions **r** are included in (C.25).

C.6.1 Lagrange-Factor and Sensitivity to Constraint

Let the bound of the constraint $c_\mu \geq 0$ in (C.10) be increased by ϵ_μ: $c_\mu \geq \epsilon_\mu$. Then the Lagrange function (C.11) becomes:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\epsilon}) = f(\mathbf{x}) - \sum_{\mu} \lambda_\mu \cdot (c_\mu(\mathbf{x}) - \epsilon_\mu) \tag{C.27}$$

From the first-order optimality condition we have that $\nabla\mathcal{L}(\mathbf{x}^*) = \mathbf{0}$ and $\nabla\mathcal{L}(\boldsymbol{\lambda}^*) = \mathbf{0}$. Therefore the sensitivity of the objective function at the minimum is identical to the sensitivity of the Lagrange function at the minimum

(C.22):

$$\nabla f^*(\epsilon_\mu) = \nabla \mathcal{L}^*(\epsilon_\mu) \overset{(C.27)}{=} \lambda_\mu^* \tag{C.28}$$

(C.28) says that the sensitivity of the minimum objective with respect to an increase in the constraint boundary is equal to the corresponding Lagrange factor.

C.7 Bounding-Box-of-Ellipsoids Property (37)

The bounding-box-of-ellipsoids property in (37) can be motivated for a parameter x_k based on the following optimization problem:

$$\max_{x_k} |x_k| \;\; \text{s.t.} \;\; \mathbf{x}^T \mathbf{C}^{-1} \mathbf{x} = \beta^2 \tag{C.29}$$

Without loss of generality, we have assumed that $\mathbf{x}_0 = \mathbf{0}$.

The first-order optimality condition (Appendix C) for a solution x_k^* of (C.29) based on a corresponding Lagrangian function,

$$\mathcal{L}(x_k, \lambda) = x_k - \frac{1}{2}\lambda \cdot (\mathbf{x}^T \mathbf{C}^{-1} \mathbf{x} - \beta^2) \tag{C.30}$$

is:

$$\nabla \mathcal{L}(\mathbf{x}^*) \equiv \mathbf{e}_k - \lambda^* \mathbf{C}^{-1} \mathbf{x}^* = \mathbf{0} \tag{C.31}$$

$$\mathbf{x}^{*T} \mathbf{C}^{-1} \mathbf{x}^* = \beta^2 \tag{C.32}$$

$\mathbf{e}_k$ is a vector with a 1 at the kth position and 0s at the other positions. Note that (C.30) can be written with any sign of λ due to the equality constraint in (C.29). Therefore, (C.30) covers both cases of (C.30), i.e., $\max x_k \equiv - \min -x_k$, and, $\min x_k$.

With the following decomposition of the covariance matrix $\mathbf{C}$,

$$\mathbf{C}^{-1} = \widehat{\mathbf{C}}^T \cdot \widehat{\mathbf{C}} \Leftrightarrow \mathbf{C} = \widehat{\mathbf{C}}^{-1} \cdot \widehat{\mathbf{C}}^{-T} \,, \tag{C.33}$$

which can be obtained by a Cholesky decomposition or an eigenvalue decomposition, (C.31) can be transformed into:

$$\widehat{\mathbf{C}} \cdot \mathbf{x}^* = \frac{1}{\lambda^*} \cdot \widehat{\mathbf{C}}^{-T} \cdot \mathbf{e}_j \tag{C.34}$$

Applying (C.33) and two times (C.34) in (C.32) leads to

$$\sigma_k = |\lambda^*| \cdot \beta \tag{C.35}$$

Applying (C.33) and one time (C.34) in (C.32) leads to

$$x_k^* = \lambda^* \cdot \beta^2 \tag{C.36}$$

From (C.35) and (C.36) and the Hessian matrix of the Lagrangian function (C.30), $\nabla^2\mathcal{L}(\mathbf{x}) = -\lambda \cdot \mathbf{C}^{-1}$, follows that the Lagrangian function (C.30) has a maximum and a minimum with the absolute value $|x_k^*| = \beta \cdot \sigma_j$.

(C.29) therefore says that any ellipsoid with any correlation value leads to a minimum and maximum value of $x_k^* = \pm\beta \cdot \sigma_k$. This shows the lub property of (37). For a complete proof we additionally have to show that in direction of other parameters all values on the bounding box are reached by varying the correlation. Instead we refer to the visual inspection of Figure 22(d).

References

[1] H. Abdel-Malek and A. Hassan. The ellipsoidal technique for design centering and region approximation. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 10:1006–1013, 1991.

[2] D. Agnew. Improved minimax optimization for circuit design. *IEEE Transactions on Circuits and Systems CAS*, 28:791–803, 1981.

[3] G. Alpaydin, S. Balkir, and G. Dundar. An evolutionary approach to automatic synthesis of high-performance analog integrated circuits. *IEEE Transactions on Evolutionary Computation*, 7(3):240–252, June 2003.

[4] Antonio R. Alvarez, Behrooz L. Abdi, Dennis L. Young, Harrison D. Weed, Jim Teplik, and Eric R. Herald. Application of statistical design and response surface methods to computer-aided VLSI device design. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 7(2):272–288, February 1988.

[5] T. Anderson. *An Introduction to Multivariate Statistical Analysis*. Wiley, New York, 1958.

[6] K. Antreich, H. Graeb, and C. Wieser. Circuit analysis and optimization driven by worst-case distances. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 13(1):57–71, January 1994.

[7] K. Antreich and S. Huss. An interactive optimization technique for the nominal design of integrated circuits. *IEEE Transactions on Circuits and Systems CAS*, 31:203–212, 1984.

[8] K. Antreich and R. Koblitz. Design centering by yield prediction. *IEEE Transactions on Circuits and Systems CAS*, 29:88–95, 1982.

[9] K. Antreich, P. Leibner, and F. Poernbacher. Nominal design of integrated circuits on circuit level by an interactive improvement method. *IEEE Transactions on Circuits and Systems CAS*, 35:1501–1511, 1988.

[10] Kurt J. Antreich, Helmut E. Graeb, and Rudolf K. Koblitz. *Advanced Yield Optimization Techniques*, Volume 8 (Statistical Approach to VLSI) of *Advances in CAD for VLSI*. Elsevier Science Publishers, Amsterdam, 1994.

[11] J. Armaos. *Zur Optimierung der Fertigungsausbeute elektrischer Schaltungen unter Beruecksichtigung der Parametertoleranzen*. PhD thesis, Technische Universitaet Muenchen, 1982.

[12] J. Bandler and S. Chen. Circuit optimization: The state of the art. *IEEE Transactions on Microwaves Theory Techniques (MTT)*, 36:424–442, 1988.

[13] J. Bandler, S. Chen, S. Daijavad, and K. Madsen. Efficient optimization with integrated gradient approximation. *IEEE Transactions on Microwaves Theory Techniques (MTT)*, 36:444–455, 1988.

[14] T. Barker. Quality engineering by design: Taguchi's philosophy. *Quality Assurance*, 13:72–80, 1987.

[15] Kamel Benboudjema, Mounir Boukadoum, Gabriel Vasilescu, and Georges Alquie. Symbolic analysis of linear microwave circuits by extension of the polynomial interpolation method. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 45(9):936, 1998.

[16] M. Bernardo, R. Buck, L. Liu, W. Nazaret, J. Sacks, and W. Welch. Integrated circuit design optimization using a sequential strategy. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 11:361–372, 1992.

[17] R. Biernacki, J. Bandler, J. Song, and Q. Zhang. Efficient quadratic approximation for statistical design. *IEEE Transactions on Circuits and Systems CAS*, 36:1449–1454, 1989.

[18] C. Borchers. Symbolic behavioral model generation of nonlinear analog circuits. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 45(10):1362, 1998.

[19] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[20] Graeme R. Boyle, Barry M Cohn, Danald O. Pederson, and James E. Solomon. Macromodeling of integrated operational amplifiers. *IEEE Journal of Solid-State Circuits SC*, 9(6):353–364, December 1974.

[21] R. Brayton, G. Hachtel, and A. Sangiovanni-Vincentelli. A survey of optimization techniques for integrated-circuit design. *Proceedings of the IEEE*, 69:1334–1363, 1981.

[22] G. Casinovi and A. Sangiovanni-Vincentelli. A macromodeling algorithm for analog circuits. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 10:150–160, 1991.

[23] R. Chadha, K. Singhal, J. Vlach, and E. Christen. WATOPT - an optimizer for circuit applications. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 6:472–479, 1987.

[24] H. Chang, E. Charbon, U. Choudhury, A. Demir, E. Felt, E. Liu, E. Malavasi, A. Sangiovanni-Vincentelli, and I. Vassiliou. *A Top-Down, Constraint-Driven Design Methodology for Analog Integrated Circuits*. Kluwer Academic Publishers, 1997.

[25] E. Christensen and J. Vlach. NETOPT – a program for multiobjective design of linear networks. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 7:567–577, 1988.

[26] M. Chu and D. J. Allstot. Elitist nondominated sorting genetic algorithm based rf ic optimizer. *IEEE Transactions on Circuits and Systems CAS*, 52(3):535–545, March 2005.

[27] L. Chua. Global optimization: a naive approach. *IEEE Transactions on Circuits and Systems CAS*, 37:966–969, 1990.

[28] Andrew R. Conn, Paula K. Coulman, Ruud A. Haring, Gregory L. Morill, Chandu Visweswariah, and Chai Wah Wu. JiffyTune: Circuit optimization using time-domain sensitivities. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 17(12):1292–1309, December 1998.

[29] P. Cox, P. Yang, S. Mahant-Shetti, and P. Chatterjee. Statistical modeling for efficient parametric yield estimation of MOS VLSI circuits. *IEEE Transactions on Electron Devices ED*, 32:471–478, 1985.

[30] Walter Daems, Georges Gielen, and Willy Sansen. Circuit simplification for the symbolic analysis of analog integrated circuits. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 21(4):395–407, April 2002.

[31] Walter Daems, Georges Gielen, and Willy Sansen. Simulation-based generation of posynomial performance models for the sizing of analog integrated circuits. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 22(5):517–534, May 2003.

[32] Walter Daems, Wim Verhaegen, Piet Wambacq, Georges Gielen, and Willy Sansen. Evaluation of error-control strategies for the linear symbolic analysis of analog integrated circuits. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 46(5):594–606, May 1999.

[33] Nader Damavandi and Safieddin Safavi-Naeini. A hybrid evolutionary programming method for circuit optimization. *IEEE Transactions on Circuits and Systems CAS*, 2005.

[34] Indraneel Das and J. E. Dennis. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization*, 8(3):631–657, August 1998.

[35] Bart De Smedt and Georges G. E. Gielen. WATSON: Design space boundary exploration and model generation for analog and RF IC design. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 22(2):213–223, February 2003.

[36] Kalyanmoy Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley-Interscience Series in Systems and Optimizat. Wiley, 2001.

[37] Maria del Mar Hershenson, Stephen P. Boyd, and Thomas H. Lee. Optimal design of a CMOS Op-Amp via geometric programming. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 20(1):1–21, January 2001.

[38] S. Director and G. Hachtel. The simplicial approximation approach to design centering. *IEEE Transactions on Circuits and Systems CAS*, 24:363–372, 1977.

[39] Alex Doboli and Ranga Vemuri. Behavioral modeling for high-level synthesis of analog and mixed-signal systems from vhdl-ams. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 2003.

[40] K. Doganis and D. Scharfetter. General optimization and extraction of IC device model parameters. *IEEE Transactions on Electron Devices ED*, 30:1219–1228, 1983.

[41] Hans Eschenauer, Juhani Koski, and Andrzej Osyczka. *Multicriteria design optimization: procedures and applications*. Springer-Verlag, 1990.

[42] Mounir Fares and Bozena Kaminska. FPAD: A fuzzy nonlinear programming approach to analog circuit design. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 14(7):785–793, July 1995.

[43] P. Feldmann and S. Director. Integrated circuit quality optimization using surface integrals. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 12:1868–1879, 1993.

[44] F. V. Fernandez, O. Guerra, J. D. Rodriguez-Garcia, and A. Rodriguez-Vazquez. Symbolic analysis of large analog integrated circuits: The numerical reference generation problem. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 45(10):1351, 1998.

[45] Roger Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, 1987.

[46] Kenneth Francken and Georges G. E. Gielen. A high-level simulation and synthesis environment for sigma delta modulators. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 22(8):1049–1061, August 2003.

[47] D.D. Gajski and R.H. Kuhn. Guest editor's introduction: New VLSI tools. *ieeecomputer*, 16:11–14, 1983.

[48] Floyd W. Gembicki and Yacov Y. Haimes. Approach to performance and sensitivity multiobjective optimization: The goal attainment method. *IEEE Transactions on Automatic Control*, 20(6):769–771, December 1975.

[49] Ian E. Getreu, Andreas D. Hadiwidjaja, and Johan M. Brinch. An integrated-circuit comperator macromodel. *IEEE Journal of Solid-State Circuits SC*, 11(6):826–833, December 1976.

[50] G. Gielen and W. Sansen. *Symbolic Analysis for Automated Design of Analog Integrated Circuits*. Kluwer Academic Publishers, Dordrecht, 1991.

[51] G. Gielen, P. Wacambacq, and W. Sansen. Symbolic analysis methods and applications for analog circuits: A tutorial overview. *Proceedings of the IEEE*, 82, 1994.

[52] G. Gielen, H. C. Walscharts, and W. C. Sansen. ISAAC: A symbolic simulation for analog integrated circuits. *IEEE Journal of Solid-State Circuits SC*, 24:1587–1597, December 1989.

[53] G. Gielen, H. C. Walscharts, and W. C. Sansen. Analog circuit design optimization based on symbolic simulation and simulated annealing. *IEEE Journal of Solid-State Circuits SC*, 25:707–713, June 1990.

[54] Georges G. E. Gielen, Kenneth Francken, Ewout Martens, and Martin Vogels. An analytical integration method for the simulation of continuous-time delta-sigma modulators. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 2004.

[55] Georges G. E. Gielen and Rob A. Rutenbar. Computer-aided design of analog and mixed-signal integrated circuits. *Proceedings of the IEEE*, 88(12):1825–1852, December 2000.

[56] Philip E. Gill, Walter Murray, and Margaret H. Wright. *Practical Optimization*. Academic Press. Inc., London, 1981.

[57] G. J. Gomez, S. H. K. Embabi, E. Sanchez-Sinencio, and M. C. Lefebvre. A nonlinear macromodel for CMOS OTAs. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, Volume 2, pages 920–923, 1995.

[58] H. Graeb, S. Zizala, J. Eckmueller, and K. Antreich. The sizing rules method for analog integrated circuit design. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 343–349, 2001.

[59] A. Groch, L. Vidigal, and S. Director. A new global optimization method for electronic circuit design. *IEEE Transactions on Circuits and Systems CAS*, 32:160–169, 1985.

[60] G. D. Hachtel and P. Zug. APLSTAP – circuit design and optimization system – user's guide. Technical report, IBM Yorktown Research Facility, Yorktown, New York, 1981.

[61] R. Hanson and C. Lawson. *Solving Least Squares Problems*. Prentice-Hall, New Jersey, 1974.

[62] R. Harjani, R. Rutenbar, and L. Carley. OASYS: A framework for analog circuit synthesis. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 8:1247–1266, 1989.

[63] D. Hocevar, P. Cox, and P. Yang. Parametric yield optimization for MOS circuit blocks. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 7:645–658, 1988.

[64] B. Hoppe, G. Neuendorf, D. Schmitt-Landsiedel, and W. Specks. Optimization of high-speed CMOS logic circuits with analytical models for signal delay, chip area, and dynamic power consumption. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 9:236–247, 1990.

[65] Xiaoling Huang, Chris S. Gathercole, and H. Alan Mantooth. Modeling nonlinear dynamics in analog circuits via root localization. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 2003.

[66] Ching-Lai Hwang and Abu Syed Md. Masud. *Multiple Objective Decision Making*. Springer, 1979.

[67] Jacob Katzenelson and Aharon Unikovski. Symbolic-numeric circuit analysis or symbolic ciruit analysis with online approximations. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 46(1):197–207, January 1999.

[68] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi. Optimization by simulated annealing. Technical Report RC 9355 (#41093), IBM Thomas J. Watson Research Center, IBM Research Division, San Jose, Yorktown, Zurich, February 1982.

[69] G. Kjellstroem and L. Taxen. Stochastic optimization in system design. *IEEE Transactions on Circuits and Systems CAS*, 28:702–715, 1981.

[70] Ken Kundert, Henry Chang, Dan Jefferies, Gilles Lamant, Enrico Malavasi, and Fred Sendig. Design of mixed-signal systems-on-a-chip. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 19(12):1561–1571, December 2000.

[71] Francky Leyn, Georges Gielen, and Willy Sansen. Analog small-signal modeling – part I: Behavioral signal path modeling for analog integrated circuits. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 48(7):701–711, July 2001.

[72] Francky Leyn, Georges Gielen, and Willy Sansen. Analog small-signal modeling – part II: Elementary transistor stages analyzed with behavioral signal path modeling. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 48(7):712–721, July 2001.

[73] M. Lightner, T. Trick, and R. Zug. Circuit optimization and design. *Circuit Analysis, Simulation and Design, Part 2 (A. Ruehli). Advances in CAD for VLSI 3*, pages 333–391, 1987.

[74] M. R. Lightner and S. W. Director. Multiple criterion optimization for the design of electronic circuits. *IEEE Transactions on Circuits and Systems CAS*, 28(3):169–179, March 1981.

[75] V. Litovski and M. Zwolinski. *VLSI Circuit Simulation and Optimization*. Chapman Hall, 1997.

[76] Hongzhou Liu, Amit Singhee, Rob A. Rutenbar, and L. Richard Carley. Remebrance of circuits past: Macromodeling by data mining in large analog design spaces. In *ACM/IEEE Design Automation Conference (DAC)*, pages 437–442, 2002.

[77] Arun N. Lokanathan and Jay B. Brockman. A methodology for concurrent process-circuit optimization. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 18(7):889–902, July 1999.

[78] K. Low and S. Director. An efficient methodology for building macromodels of IC fabrication processes. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 8:1299–1313, 1989.

[79] D. Luenberger. *Optimization By Vector Space Methods*. John Wiley, New York, 1969.

[80] David G. Luenberger. *Linear and Nonlinear Programming*. Addison-Wesley Publishing Company, 2 edition, May 1989.

[81] Pradip Mandal and V. Visvanathan. CMOS Op-Amp sizing using a geometric programming formulation. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 20(1):22–38, January 2001.

[82] H. Alan Mantooth and Mike F. Fiegenbaum. *Modeling with an Analog Hardware Description Language*. Kluwer Academic Publishers, November 1994.

[83] P. Maulik, L. R. Carley, and R. Rutenbar. Integer programming based topology selection of cell-level analog circuits. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 14(4):401ff, April 1995.

[84] Petra Michel, Ulrich Lauther, and Peter Duzy. *The Synthesis Approach to Digital System Design*. Kluwer Academic Publishers, Boston, 1992.

[85] Gordon E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38(8), April 1965.

[86] Daniel Mueller, Guido Stehr, Helmut Graeb, and Ulf Schlichtmann. Deterministic approaches to analog performance space exploration (PSE). In *ACM/IEEE Design Automation Conference (DAC)*, June 2005.

[87] G. Mueller-Liebler. PASTA – The characterization of the inherent fluctuations in the fabrication process for circuit simulation. *International Journal of Circuit Theory and Applications*, 23:413–432, 1995.

[88] MunEDA. *WiCkeD – Design for Manufacturability and Yield*. www.muneda.com, 2001.

[89] L. Nagel. *SPICE2: A computer program to simulate semiconductor circuits*. Ph. D. dissertation, Univ. of California, Berkeley, 1975.

[90] Dongkyung Nam and Cheol Hoon Park. Multiobjective simulated annealing: A comparative study to evolutionary algorithms. *International Journal of Fuzzy Systems*, pages 87–97, June 2000.

[91] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 1999.

[92] W. Nye, D. Riley, A. Sangiovanni-Vincentelli, and A. Tits. DELIGHT.SPICE: An optimization-based system for the design of integrated circuits. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 7:501–519, 1988.

[93] E. Ochotta, T. Mukherjee, R.A. Rutenbar, and L.R. Carley. *Practical Synthesis of High-Performance Analog Circuits*. Kluwer Academic Publishers, 1998.

[94] Emil S. Ochotta, Rob A. Rutenbar, and L. Richard Carley. Synthesis of high-performance analog circuits in ASTRX/OBLX. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 15(3):273–294, March 1996.

[95] M. Pelgrom, A. Duinmaijer, and A. Welbers. Matching properties of MOS transistors. *IEEE Journal of Solid-State Circuits SC*, 24:1433–1440, 1989.

[96] Rodney Phelps, Michael Krasnicki, Rob A. Rutenbar, L. Richard Carley, and James R. Hellums. ANACONDA: Simulation-based synthesis of analog circuits via stochastic pattern search. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 19(6):703–717, June 2000.

[97] Lawrence T. Pillage, Ronald A. Rohrer, and Chandramouli Visweswariah. *Electronic Circuit and System Simulation Methods*. McGraw-Hill, Inc., 1995.

[98] Ming Qu and M. A. Styblinski. Parameter extraction for statistical ic modeling based on recursive inverse approximation. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 16(11):1250–1259, 1997.

[99] Joao Ramos, Kenneth Francken, Georges G. E. Gielen, and Michiel S. J. Steyaert. An efficient, fully parasitic-aware power amplifier design optimization tool. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 2005.

[100] Carl R. C. De Ranter, Geert Van der Plas, Michiel S. J. Steyaert, Georges G. E. Gielen, and Willy M. C. Sansen. CYCLONE: Automated design and layout of RF LC-oscillators. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 21(10):1161–1170, October 2002.

[101] A. Ruehli(Editor). *Circuit Analysis, Simulation and Design*. Advances in CAD for VLSI. North-Holland, 1986.

[102] Youssef G. Saab and Vasant B. Rao. Combinatorial optimization by stochastic evolution. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 10(4):525–535, April 1991.

[103] T. Sakurai, B. Lin, and A. Newton. Fast simulated diffusion: an optimization algorithm for multiminimum problems and its application to MOSFET model parameter extraction. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 11:228–233, 1992.

[104] Sachin S. Sapatnekar, Vasant B. Rao, Pravin M. Vaidya, and Sung-Mo Kang. An exact solution to the transistor sizing problem for CMOS circuits using convex optimization. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 12(11):1621–1634, November 1993.

[105] M. Sharma and N. Arora. OPTIMA: A nonlinear model parameter extraction program with statistical confidence region algorithms. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 12:982–987, 1993.

[106] C.-J. Richard Shi and Xiang-Dong Tan. Canonical symbolic analysis of large analog circuits with determinant decision diagrams. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 19(1):1–18, January 2000.

[107] C.-J. Richard Shi and Xiang-Dong Tan. Compact representation and efficient generation of s-expanded symbolic network functions for computer-aided analog circuit design. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 20(7):813, July 2001.

[108] Guoyong Shi, Bo Hu, and C.-J. Richard Shi. On symbolic model order reduction. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 2006.

[109] C. Spanos and S. Director. Parameter extraction for statistical IC process characterization. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 5:66–78, 1986.

[110] Thanwa Sripramong and Christofer Toumazou. The invention of cmos amplifiers using genetic programming and current-flow analysis. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 2002.

[111] H.H. Szu and R.L. Hartley. Nonconvex optimization by fast simulated annealing. *Proceedings of the IEEE*, 75:1538–1540, 1987.

[112] Sheldon X.-D. Tan. A general hierarchical circuit modeling and simulation algorithm. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 2005.

[113] Sheldon X.-D. Tan and C.-J. Richard Shi. Efficient approximation of symbolic expressions for analog behavioral modeling and analysis. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 2004.

[114] Xiangdong Tan and C.-J. Richard Shi. Hierarchical symbolic analysis of large analog circuits with determinant decision diagrams. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, page VI/318, 1998.

[115] Hua Tang, Hui Zhang, and Alex Doboli. Refinement-based synthesis of continuous-time analog filters through successive domain pruning, plateau search, and adaptive sampling. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 2006.

[116] Antonio Torralba, Jorge Chavez, and Leopoldo G. Franquelo. FASY: A fuzzy-logic based tool for analog synthesis. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 15(7):705–715, July 1996.

[117] Wim M. G. van Bokhoven and Domine M. W. Leenaerts. Explicit formulas for the solutions of piecewise linear networks. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 46(9):1110ff., September 1999.

[118] Geert Van der Plas, Geert Debyser, Francky Leyn, Koen Lampaert, Jan Vandenbussche, Georges Gielen, Willy Sansen, Petar Veselinovic, and Domine Leenaerts. AMGIE–A synthesis environment for CMOS analog integrated circuits. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 20(9):1037–1058, September 2001.

[119] P. Wambacq, P. Dobrovolny, G. G. E. Gielen, and W. Sansen. Symbolic analysis of large analog circuits using a sensitivity-driven enumeration of common spanning trees. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 45(10):1341, 1998.

[120] P. Wambacq, G. G. E. Gielen, and W. Sansen. Symbolic network analysis methods for practical analog integrated circuits: A survey. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 45(10):1331, 1998.

[121] Jacob K. White and Alberto Sangiovanni-Vincentelli. *Relaxation Techniques for the Simulation of VLSI Circuits*. Kluwer Academic Publishers, 1987.

[122] Claudia Wieser. *Schaltkreisanalyse mit Worst-Case Abstaenden*. PhD thesis, Technische Universitaet Muenchen, 1994.

[123] J. Wojciechowski and J. Vlach. Ellipsoidal method for design centering and yield estimation. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 12:1570–1579, 1993.

[124] X. Xiangming and R. Spence. Trade-off prediction and circuit performance optimization using a second-order model. *International Journal of Circuit Theory and Applications*, 20:299–307, 1992.

[125] D. Young, J. Teplik, H. Weed, N. Tracht, and A. Alvarez. Application of statistical design and response surface methods to computer-aided VLSI device design II: desirability functions and Taguchi methods. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 10:103–115, 1991.

[126] T. Yu, S. Kang, I. Hajj, and T. Trick. Statistical performance modeling and parametric yield estimation of MOS VLSI. *IEEE Transactions on Computer-Aided Design of Circuits and Systems*, 6:1013–1022, 1987.

[127] T. Yu, S. Kang, J. Sacks, and W. Welch. Parametric yield optimization of CMOS analogue circuits by quadratic statistical circuit performance models. *International Journal of Circuit Theory and Applications*, 19:579–592, 1991.

[128] J. Zou, D. Mueller, H. Graeb, and U. Schlichtmann. A CPPLL hierarchical optimization methodology considering jitter, power and locking time. In *ACM/IEEE Design Automation Conference (DAC)*, pages 19–24, 2006.

Index

- Acceptance function, 75, 107, 145
- Acceptance region
  - parameter, 73
  - performance, 48, 72
- AC simulation, 46
- Analog circuit, 4
- Analog design, 6
- Analog sizing, 10, 12
- Analog synthesis, 9
  - parametric, 10–11
  - path, 12
  - structural, 10–11
- Architecture level, 11
- Center of gravity, 147
- Central moment, 166
- χ^2(chi-square)-distribution, 42, 120
- Circuit level, 11
- Circuit netlist, 8
- Condition number, 55
- Confidence level, 114
- Corner worst case, 90
  - excessive robustness, 127
- Correlation, 35, 166
- Covariance matrix, 35, 146, 166
  - linear transformation, 167
- Cumulative distribution function, 31
- Cumulative frequency function, 31
- DC simulation, 46
- Descent direction, 174, 177
- Design centering, 13, 76
- Design flow, 10
- Design level, 6
- Design partitioning, 7
- Design technology, 2
- Design view, 8
- Digital design, 6
- Discrete parameter, 89
- Error function, 33
- Estimator, 108
  - bias, 170
  - variance, 170
- Expectation value, 32, 75, 108, 165
  - estimator, 169
    - variance, 170
  - linear transformation, 167
- First-order optimality condition
  - classical worst-case analysis, 87
  - constrained optimization, 179
  - general worst-case analysis, 97
  - geometric yield analysis, 132
  - realistic worst-case analysis, 92
  - statistical-yield optimization, 147
  - unconstrained optimization, 174
- Gaussian error propagation, 167
- Gauss-Newton direction, 159
- Geometric yield analysis, 14, 128
  - accuracy, 141
  - complexity, 141
- Geometric-yield optimization, 14, 153, 156
  - least-squares/trust-region, 158
  - linear programming, 162
  - min-max, 162
- Gradient, 16, 50
  - statistical-yield optimization, 146
    - deterministic design parameter, 152
  - worst-case distance, 154
- Hessian matrix
  - statistical-yield optimization, 148
- Importance sampling, 110
- Jacobian matrix, 50
- Karush-Kuhn-Tucker(KKT) conditions, 179
- Lagrangian function
  - classical worst-case analysis, 87
  - general worst-case analysis, 97
  - geometric yield analysis, 131
  - realistic worst-case analysis, 92
- Least-squares optimization, 159
- Levenberg-Marquardt direction, 159
- Linear programming, 87, 162

Lognormal distribution, 41
Macromodels, 47
Manufacturing tolerances, 13
Mean value, 32, 146, 165
Min-max optimization, 162
Mismatch, 45
Mixed-signal circuit, 5
Moment, 165
Monte-Carlo analysis, 14, 109
  accuracy, 115
  complexity, 115
Multiple-criteria optimization, 56
Multiple-objective optimization, 15, 56, 69, 77, 156
Multivariate normal distribution, 34
Newton-type optimization, 150
Node voltages, 46
Nominal design, 13, 20, 56, 77, 162
Numerical integration, 46
Numerical simulation, 16, 46, 69, 78, 108
Operating tolerances, 13
Operational amplifier, 4, 8, 16, 156
Parameter, 27
  design parameter, 27, 78
  range parameter, 28
  significance, 51
  similarity, 51
  statistical parameter, 27
  tolerances, 28–29
    global, 13, 44
    local, 13, 44
Pareto front, 59, 159
Pareto optimality, 57
Pareto optimization, 61
Pareto point, 59
Performance feature, 45, 51–52
Performance-feature bound, 48, 63, 65, 86, 91, 95
Performance-specification feature, 75
Performance specification, 47, 70
Performance-specification feature, 47, 73, 108, 121, 129
Phase-locked loop, 5, 10
Probability density function, 22, 31, 37, 71
  truncated, 145
Random variable, 32
  standardization, 168
RC circuit, 19, 105, 143
  nominal design, 20
  performance specification, 20
  yield optimization/design centering, 22
Relative frequency function, 31
Response surface modeling, 47
Sample, 108
  element, 169
  generation, 43
  size, 114
Scaling, 54
  affine transformation, 55
  covered range, 54
  reference point, 54
  sensitivity, 55
Second-order optimality condition
  constrained optimization, 180
  general worst-case analysis, 97
  geometric yield analysis, 133
  realistic worst-case analysis, 92
  statistical-yield optimization, 149
  unconstrained optimization, 175
Sensitivity, 49
  computation, 53, 69
  finite-difference approximation, 53, 150
  Lagrange factor, 180
  matrix, 50
Sequential Quadratic Programming, 97, 131
Single-objective optimization, 15, 61, 76, 162
Singular distribution, 36
Six-sigma design, 94, 100
Smoothness, 49
Standard deviation, 32, 35, 166
Standard normal distribution, 33
Statistical yield analysis, 14
Statistical-yield optimization, 14, 150
Symbolic analysis, 47
Three-sigma design, 94, 100
Tolerance assignment, 80, 150
Tolerance class, 115
  box, 29, 118
  ellipsoid, 29, 119
  interval, 116
  polytope, 29
  single-plane-bounded, 122, 128
Tolerance design, 13
Trade-off, 57
Transformation of statistical distributions, 40
TR simulation, 46
Truncated distribution, 89
Trust-region optimization, 159
Unconstrained direction, 177
Unconstrained optimization, 173
Uniform distribution, 41
Univariate normal distribution, 32
Variance, 33, 166
Variance/covariance matrix, 35, 166
  estimator, 169
Vector norm, 63
  l_1, 63
  l_2, 63
  l_∞, 59, 63
Worst-case analysis, 64, 66, 69, 76
  classical, 85–86
  general, 95
  realistic, 85, 90, 105
Worst-case distance, 94, 100, 131, 136, 156
  target, 158
Worst-case optimization, 13, 64, 68, 77

Worst-case parameter vector, 57, 65–66
classical, 88
general, 98
geometric yield analysis, 135
realistic, 93
Worst-case performance, 65–66
classical, 89
general, 98
realistic, 93
Y-chart, 8
Yield, 65, 70, 72, 108
Yield analysis, 75, 78
Yield approximation error, 103
Yield
"cake", 25
catastrophic, 23
Yield estimator, 108
variance, 111
Yield optimization, 13, 76
Yield
parametric, 23
Yield partition, 75, 108, 121, 137
Z-branch currents, 46