SPRINGER BRIEFS IN PHYSICS

Edmund J. Sullivan

Model-Based Processing for Underwater Acoustic Arrays

ASA Press

Springer

SpringerBriefs in Physics

More information about this series at http://www.springer.com/series/8902

Edmund J. Sullivan

Model-Based Processing for Underwater Acoustic Arrays

ASA Press

Springer

Edmund J. Sullivan
Portsmouth, RI, USA

ISSN 2191-5423 ISSN 2191-5431 (electronic)
SpringerBriefs in Physics
ISBN 978-3-319-17556-0 ISBN 978-3-319-17557-7 (eBook)
DOI 10.1007/978-3-319-17557-7

Library of Congress Control Number: 2015938454

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media (www.springer.com)

To Lori, who left us far too soon

Preface

This monograph presents a reasonably complete treatment of the model-based approach to the processing of data from underwater acoustic arrays. By complete we mean that the material herein is accessible to anyone at the level of a bachelor's degree in engineering, but may not be sufficiently familiar with the areas of statistical signal processing or acoustic array processing. With this goal in mind, it provides a reasonably rigorous treatment of standard time-domain statistical signal processing and acoustic array processing with an emphasis on its spatial processing aspects. A second reason for taking this approach is that since the processing philosophy presented here differs sufficiently from the standard approach, a review of the standard approach was warranted.

At its heart, model-based processing as discussed here is a form of Bayesian processing that relies heavily on physical models to provide the a priori information. This is done within the framework of the Kalman filter, since it itself is a Bayesian processor, and additionally provides a natural framework for including physical models, along with the ability of including prior information in a statistical form as in the usual Bayesian processor. Further, it is capable of easily handling the nonlinearities that accompany real-world models. By physical models we mean here such phenomena as array motion, array configuration, signal structures other than plane wave models, and oceanic propagation models. Because of this, the material presented here constitutes an approach to acoustic array processing that is capable of providing performance improvement over many of the presently used methods.

I would like to acknowledge Dr. James Candy, Chief Scientist for Engineering, the Lawrence Livermore National Laboratory for originally introducing me to the Kalman filter and emphasizing its applicability far beyond its original area of control theory. He made it clear that it provides a framework for performance enhancement to the field of signal processing and estimation theory in a very general way. I also gratefully acknowledge Dr. Allan Pierce, Professor Emeritus of Mechanical Engineering, Boston University, who encouraged me to write this book.

Portsmouth, RI, USA — Edmund J. Sullivan
December 2014

Contents

Chapter 1
Introduction

1.1 Background

The underwater acoustic array is generally concerned with the tasks of detection, estimation, and tracking. Here, detection is defined as the determination of the existence or nonexistence of a postulated signal at the receiver, and in its simplest form it is a simple binary (yes/no) decision. Once this decision is made, the task of detection is complete. The next step is estimation, where the signal is examined in order to obtain more information. When the estimation is concerned with the target coordinates, then it is called tracking. This book is mainly concerned with estimation, and detection per se is not discussed in great detail.

For our purposes, it is convenient to divide estimation into two classes: parametric and nonparametric. What is meant by nonparametric estimation is that the processing does not concern itself with other than the value of the estimate. Conventional bearing estimation using a line array would be a simple example of a nonparametric estimator. Parametric, on the other hand, attempts to exploit certain parameters of the signal, as in time series analysis, or parameters describing the source and the medium. These ideas are depicted in Fig. 1.1. Here it is seen that one kind of parametric processing is the *inverse problem*, which is a means of extracting information from the data regarding the source or the medium.

1.2 The Inverse Problem

Estimation is the determination of the values of certain parameters of the signal, the source, or the medium. As mentioned above, a simple example is the determination of the bearing angle of the signal at a receiving array of hydrophones arriving from an acoustic source. At its most complex, estimation leads to a class of

E.J. Sullivan, *Model-Based Processing for Underwater Acoustic Arrays*,
SpringerBriefs in Physics, DOI 10.1007/978-3-319-17557-7_1

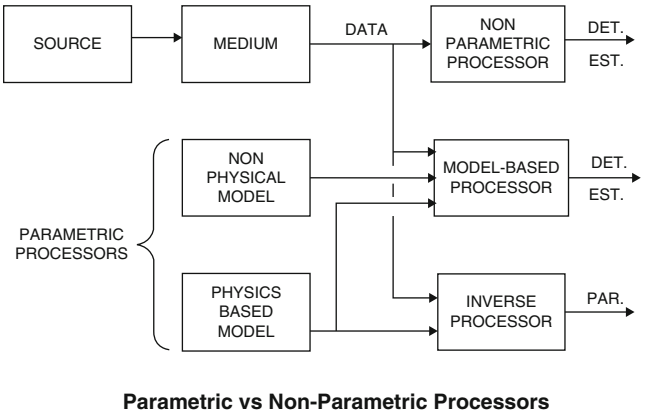

Fig. 1.1 Classification of processors

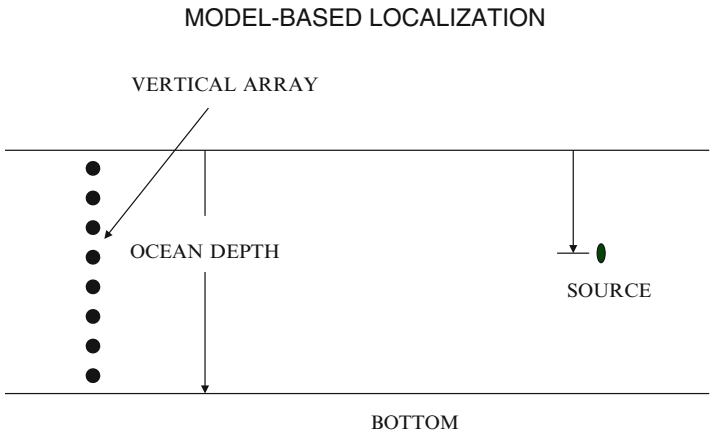

Fig. 1.2 Ideal oceanic waveguide

problems called *inverse problems*, examples of which could be source localization, the extraction of the value of some property of the ocean (sound speed), the ocean bottom (density, sound speed, etc.) from the signal, or the characterization of a source or scatterer, commonly called identification.

One way to look at the inverse problem is to define it in terms of an associated forward problem. Consider an acoustic source in an ideal two-dimensional oceanic waveguide as depicted in Fig. 1.2. The surface and bottom are the boundaries of the waveguide. If the source frequency, ocean depth, bottom characteristics and the range from the source to the vertical array receiver and the depth of the source are all known, and an appropriate propagation model is available, then the acoustic field at the array elements can be computed. This is called the *forward* problem. An associated inverse problem would be the following. Given the propagation model,

source frequency, ocean depth, bottom characteristics, and the array measurements, what are the coordinates (range and depth) of the source? This is the so-called matched field (MF) problem of ocean acoustics and was first examined by Hinich [9] in 1972, where the depth-only problem was examined, and later in 1976 by Bucker [3] who showed that both the range and depth of the source could be estimated. Bucker introduced the term "matched-field processing" or MFP.

Since the work of Hinich and Bucker, a great deal of work has been done on MFP and it has proven to be quite useful in many applications, especially those where the propagation model was well known. However, when the model was not well known, the technique would often fail. An example of this can be seen in the work of Hinich and Sullivan [10], where an experimental determination of the range and depth of a 190 Hz source was carried out using MFP with a normal-mode model used to describe the propagation. The signal-to-noise ratio (SNR) was quite high and the bottom properties were well known. There were nine modes induced by the signal. However, it was impossible to obtain a solution unless the two highest modes were removed. That is, these modes were introducing unacceptable errors into the solution which did not simply degrade the solution, but prevented any solution at all.

Ultimately, this problem constituted a practical limit on the matched-field problem. Much of the difficulty lies with the fact that the relationship between the data and the model parameters is highly nonlinear one, so that a useful quantitative measure of the limitations of the model is not available. An overview of MFP can be found in [2, 15] and references therein.

Thus, the general inverse problem contains two pitfalls. First, it is not always clear that there is sufficient information available for a satisfactory solution. For example, in the matched-field problem, the sound speed profile in the ocean may not be sufficiently well known. Second, if knowledge of the sound speed profile is necessary, then it must be known to an unknown level of accuracy. If the solution is not strongly dependent on its accuracy, it can be approximate and a usable solution can be found. Alternatively, if it must be known accurately and sufficiently accurate information is not available, the problem is considered to be *ill-posed*.[1] This can have effects ranging from a poor solution to a catastrophic loss of any solution at all. In the matched field literature, this issue has historically been called the *Mismatch* problem [16]. There have been many attempts to remedy the mismatch problem, the first major step forward being the work of Richardson and Nolte [14], who included a priori probabilities for the troublesome parameters in their Matched Field Problem algorithm to account for their uncertainties. Nevertheless, the mismatch problem remains a major limiting factor to the matched-field problem. A more detailed description of the history and methods of MFP can be found in [6] which is a special issue of *The Journal of Oceanic Engineering* dedicated to the field.

[1] A problem is considered to be ill-posed if a unique solution does not exist. Further, if it does exist, it must change continuously with changes in the initial conditions. This is sometimes referred to as being ill-posed in the sense of Hadamard [8].

In order to move beyond the limitations posed by the mismatch problem, it is necessary to look at what MFP is trying to do in a more formal sense. When one introduces a model into an estimation procedure, as in MFP, it really constitutes the use of a priori information. That is, it becomes a Bayesian problem. Strictly speaking, much of Bayesian estimation theory is based on introducing the prior information in terms of probability distributions. A classic example of this is the maximum a posteriori (MAP) [12] method.

The MAP method is a generalization of the maximum likelihood (ML) estimator. The ML estimator requires knowledge of the likelihood function, which is the probability of the measurement conditioned on the unknown parameters. Consider the case of a single unknown parameter. The likelihood function is written as

$$L = p(y|x), \tag{1.1}$$

where the measurement y is conditioned on the unknown parameter x. The ML estimate of x is then found as that value of x that maximizes $p(y|x)$. The quality of the estimate is embodied in the variance on the estimate, signified by σ^2_{ml} which is defined as the expected value of the square of the estimate minus its true value. Now suppose that there exists prior knowledge of x in terms of the probability density function $p(x)$ with variance σ^2_p. The MAP estimate is then found from the maximum of $p(y|x)p(x)$. As we will see in Chap. 3, the variance on the estimate is now given by

$$\sigma^2_{\text{map}} = \frac{\sigma^2_{\text{ml}}\sigma^2_p}{\sigma^2_{\text{ml}} + \sigma^2_p}. \tag{1.2}$$

It is of interest to rewrite this as

$$\frac{1}{\sigma^2_{\text{map}}} = \frac{1}{\sigma^2_{\text{ml}}} + \frac{1}{\sigma^2_p}. \tag{1.3}$$

The reciprocal of the variance is called the Fisher information, and for more than one parameter, it is a matrix [12]. Equation 1.3 states that the Fisher information in the MAP processor is the sum of the Fisher information[2] in the ML estimate plus that added by the prior.

1.3 Model-Based Processing

In the case of the MAP estimator, it is clear that the addition of prior information improves performance. In the case of prior information in the form of physical models however, what is the analogous procedure? As will be seen, this problem can

[2]Strictly speaking, the Fisher information is the correct term only when it derives from the likelihood function. Here it is being used in a looser sense following the usage of Frieden [7].

be dealt with in a fundamental way by embedding the physical model into a Kalman filter.[3] The Kalman filter is a Bayesian processor and is based on a Gauss–Markov model [5]. It is a recursive processor that is self-correcting since it uses new data to update the estimate. Furthermore, as will be shown in Chap. 4, it provides a means of including the poorly known parameters as part of the state vector of unknowns; a procedure known as *augmentation*. This is a powerful technique, since if these troublesome parameters can impact the estimate, then they must impact the data and are therefore amenable to estimation themselves. In other words, the Kalman filter can provide an estimate of any parameter that is *observable*, where a necessary (but not sufficient) condition for observability is that the relevant information be contained in the data. It is this approach to the problem that we refer to here as model-based processing or MBP, and is the basis of the approach discussed in this book.

Although the Kalman filter and its variants will be discussed in Sect. 4, it is useful to mention here that it provides a great deal of latitude in the fidelity of the model. It does this by allowing what is called *system* or *plant* noise, which is a critical part of the algorithm. It is not a noise in the same sense as measurement noise[4] but nevertheless it is a valid means of allowing for model deficiencies in terms of a Markov process, and it plays a critical role in the algorithm by allowing model errors to be included in a fundamental way. By model deficiencies, we mean that the model does not contain the physics in a sufficiently *complete* way. This differs from mismatch, which can exist in models that are not deficient, but contains poorly known parameters. The salient point here is that mismatch can introduce corrupting and sometimes fatal misinformation to the algorithm, whereas model deficiency simply means a lack of relevant information and can often be compensated for by the system noise, albeit with a loss in solution quality.

As is well known, most problems of interest are nonlinear, where the Kalman filter in its original form is only optimal for the linear and Gaussian case. This has not proven to be a major issue since the nonlinear problem is easily dealt with by the extended Kalman filter (EKF) [4] and the unscented Kalman filter (UKF) [11]. A more recent form of the Kalman-type processor is the particle filter [1], which allows for the case of multi-modal pdf's. Its application to the oceanic problem has been pioneered by Candy [5], Michelopolou [13], and Yardim et al. [17].

[3] The introduction of the Kalman filter into the oceanic inverse problem is due mainly to Candy [4] who pointed out that in principle, any physical model could be embedded into it.

[4] It differs from measurement noise in that it does not arise as an additive corruption of a measurement, but is a statistical measure of the lack of fidelity of the model. Nevertheless it can be considered to be a valid noise term in the Gauss–Markov sense.

1.4 Observability

Observability has two requirements. First, the relevant information must be contained in the measurement data. And second, the processor must be constructed so as to extract this information. A good example is that of wavefront curvature range estimation. If the ability to sense the wavefront curvature is not specifically modeled in the processor, array length and SNR notwithstanding, a range estimation is simply not achievable even though the range information is contained in the data.

The concept of observability [4] comes from the control theory community. In its most general form, it requires the ability of recovering the initial state of a system from measurements on the present state and can be cast in levels of increasing complexity, but for our purposes the following will suffice. Given the parameter vector $\mathbf{x}$ and the linear measurement system

$$y(t) = C\mathbf{x}, \tag{1.4}$$

observability simply means that C is an invertible matrix, since this allows the reconstruction of the initial value of $\mathbf{x}$. More generally, for a nonlinear measurement system, which is generally the case in MBP, we would have

$$y(t) = c[\mathbf{x}], \tag{1.5}$$

with $c[\mathbf{x}]$ denoting the nonlinear function of the state or parameter vector. Here, observability implies the invertibility of this function.

As an example, consider a horizontal line array of N uniformly shaded (weighted) elements, equally spaced at distance d, with a narrowband plane wave signal from a localized[5] source arriving at angle θ as measured from broadside. If a conventional beamformer is used, the phase shift of the output of each receiver element is found, from which the bearing angle θ can be estimated. However, since the signal is arriving from a localized source, and the receiving array is not far from this source, then it can be assumed that the wavefront is the arc of a circle, with the source range being the radius of this circle. Hence the range is observable. However, the conventional beamformer cannot estimate this range, since the geometry of the situation is not modeled into the measurement system. If this is done, the range and bearing can both be estimated, and in both cases, the performance of the estimators depends upon such parameters as L/λ, the aperture in units of wavelength, and R/L, the ratio of the range to the aperture.

In this example, the range and bearing are both observable (contained in the data), but their estimates depend upon the models used by the estimator. On other hand, the elevation angle, that is, the angle made by the range with the plane defined by the range and bearing, is not observable since the relevant information is not

[5] By localized we mean small enough to be approximated by a point.

contained in the data. This is a consequence of the axial symmetry of the line array and therefore no model exists that will allow its estimation. A different receiver array would be necessary in order to capture the information.

1.5 Book Outline

The remainder of the book is organized as follows. The next chapter presents an overview of the acoustic array and the conventional analysis methods used. This serves two purposes: to familiarize the acoustic array to those whose knowledge of the area is limited, and to provide a background for the newer, model-based approach in order to emphasize the significant differences involved. Chapter 3 presents a comprehensive overview of statistical signal processing as a preparation for the introduction of the Kalman-type processor in Chap. 4. Chapter 5 contains several examples of Model-Based Array Processing using both real and synthesized data. In Chap. 6, the issues of "tuning" the filter and determining the quality of the solution are discussed.

References

1. Arulampalam, S., Maskell, S., Gordon, N., Clapp, T.: A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Trans. Signal Process. **50**(2), 174–188 (2004)
2. Baggeroer, A.B., Kuperman, W.A., Schmidt, H.: Matched-field processing: source localization in correlated noise as an optimum parameter estimation problem. J. Acoust. Soc Am. **82**, 606–613 (1987)
3. Bucker, H.P.: Use of calculated sound fields and matched-field detection to locate sound sources in shallow water. J. Acoust. Soc. Am. **59**, 368–372 (1976)
4. Candy, J.V.: Model-Based Processing. Detection, Estimation, and Time-series Analysis. IEEE Press/Wiley, New York (2005)
5. Candy, J.V.: Bayesian Signal Processing: Classical, Modern, and Particle Filtering Methods. IEEE Press/Wiley, New York (2009)
6. Doolittle, R.E., Tolstoy, A., Sullivan, E.J. (eds.): Special issue on matched-field processing. IEEE Trans. Ocean. Eng. **18** (1993)
7. Frieden, B.R.: Physics from Fisher Information. Cambridge University Press, Cambridge (1998)
8. Parker, S.B. (ed.): McGraw-Hill Dictionary of Scientific and Technical Terms, 4th edn. McGraw-Hill, New York (1989)
9. Hinich, M.J.: Maximum likelihood processing for a vertical array. J. Acoust. Soc. Am. **54**, 499–503 (1972)
10. Hinich, M.J., Sullivan, E.J.: Maximum likelihood passive localization using mode filtering. J. Acoust. Soc. Am. **85**, 2214–2219 (1989)
11. Julier, S., Uhlmann, J.: Unscented filtering and nonlinear estimation. Proc. IEEE **92**, 401–422 (2004)
12. Kay, S.M.: Fundamentals of Statistical Signal Processing; Estimation Theory. Addison-Wesley, Reading (1991)

13. Michalopoulou, Z.-H., Ma, X.: Source localization in the Haro Strait primer experiment using arrival time estimation and linearization. J. Acoust. Soc. Am. **118**, 2924–2933 (2005)
14. Richardson, A.M.: *a posteriori* probability source localization in an uncertain sound speed, deep ocean environment. J. Acoust. Soc. Am. **89**, 2280–2284 (1991)
15. Sullivan, E.J., Middletom, D.: Estimation and detection issues in matched-field processing. IEEE Trans. Ocean. Eng. **18**, 156–167 (1993)
16. Tolstoy, A.: Matched-Field Processing for Ocean Acoustics. World Scientific Publishing Co., River Edge (1993)
17. Yardim, C., Gerstoft, P., Hodgekiss, W.S.: Tracking of geoacoustic parameters using Kalman and particle filters. J. Acoust. Soc. Am. **125**, 746–760 (2009)

Chapter 2
The Acoustic Array

2.1 The Acoustic Array

Hydrophones configured in groups are called acoustic arrays. An acoustic array provides a means of controlling the directional properties of acoustic transmission and reception. Here, we shall consider only receiving arrays, since most transmitting arrays[1] play a minor role in Model-Based Processing as discussed in this book.

When discussing receiving arrays, the hydrophones are often referred to as *elements* or *receivers*. The simplest configuration for an array is that where the elements are equally spaced in a line where all of the receivers are the same and have the same isotropic sensitivity. Isotropic means that the receiver is equally sensitive in all spatial directions. Much of this chapter will concentrate on the line array, since the towed array, which is a line array, is a major workhorse of naval sonar systems. Also, most of the issues surrounding the fundamental characteristics of arrays can be treated in the context of the line array.

Other forms of arrays are planar (two-dimensional) and nonplanar such as hull-mounted arrays which can be cylindrical or spherical in shape. They differ from the line array not only in performance but also in the approaches used in their analysis. These are generally referred to as three-dimensional arrays and will also be discussed in this chapter.

[1] A transmit arrays differs from a receive array in that the intensity of the transmitted field can cause a mutual coupling between its transducer elements that can strongly impact its performance. This phenomenon complicates the analysis of such arrays.

E.J. Sullivan, *Model-Based Processing for Underwater Acoustic Arrays*,
SpringerBriefs in Physics, DOI 10.1007/978-3-319-17557-7_2

2.2 The Line Array

In the case of uniformly spaced receiver elements, the length of a line array is given by

$$L = (N-1)d, \tag{2.1}$$

where L is referred to as the aperture. N is the number of elements in the array and d is the element spacing. The approximation $L \approx Nd$ holds for most calculations when N is on the order of 10 or greater. A line array is shown schematically in Fig. 2.1. In this figure, θ is the angle of the propagation direction of the incoming plane acoustic wave measured clockwise from the normal to the array or the "broadside" direction. The elemental outputs are voltages that are proportional to the instantaneous acoustic signal (pressure). In the conventional array processor, the signals from the hydrophones are weighted and summed, producing the array output. For unity weighting, this summation is described by

$$S = \sum_{n=0}^{N-1} s_n(t). \tag{2.2}$$

And assuming a narrowband signal with radian frequency $\omega = 2\pi f$ where f is the frequency, the signal at the nth receiver can be written in complex form, as

$$s_n = e^{i\omega(t+\tau_n)} \tag{2.3}$$

so that the array output, S, is given by

$$S = e^{i\omega t} \sum_{n=0}^{N-1} e^{i\omega\tau_n}. \tag{2.4}$$

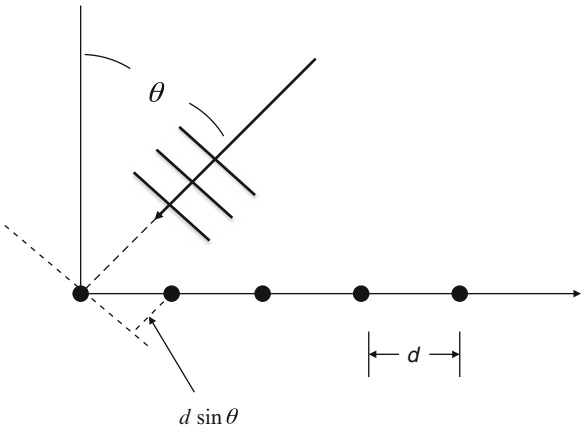

Fig. 2.1 Line array configuration

In this equation, τ_n is the time delay of the wavefront associated with the nth receiver. With the help of Fig. 2.1 we see that, by defining the delay at the first ($n = 0$) receiver to be zero, and recognizing that the wavefront is normal to the propagation direction, the relative delay associated with the second receiver is the travel time τ associated with the distance $d\sin\theta$, that is $\tau = (d/c)\sin\theta$, where c is the speed of sound in water, so that for the nth receiver the time delay is given by τ_n where

$$\tau_n = n\tau = n(d/c)\sin\theta. \tag{2.5}$$

Note that the time dependence was factored out of the sum in Eq. 2.4. This allows us to consider the array output in terms of its spatial processing performance only. In the case of a single-frequency signal such as we have here, we choose to write the time delay in terms of its associated phase shift, since this will be important later. Note that the phase associated with the time delay τ_n is given by $\omega\tau_n$, so that the phase associated with the nth element, which we call ϕ_n, is given by

$$\phi_n = \omega\tau_n = nd(\omega/c)\sin\theta. \tag{2.6}$$

For the single frequency case, it is convenient to define a spatial frequency or *wavenumber* $k = \omega/c = 2\pi f/c = 2\pi/\lambda$, with λ being the wavelength. With this definition, the phase at the nth receiver is now written as

$$\phi_n = nkd\sin\theta. \tag{2.7}$$

Now writing Eq. 2.4 explicitly as a function of θ and without the time dependent part, the spatial part becomes

$$S(\theta) = \sum_{n=0}^{N-1} e^{inkd\sin\theta}, \tag{2.8}$$

or generally, if the individual elements have a directivity $f(\theta)$, this becomes

$$S(\theta) = \sum_{n=0}^{N-1} f(\theta)e^{inkd\sin\theta} = f(\theta)\sum_{n=0}^{N-1} e^{inkd\sin\theta}. \tag{2.9}$$

In this case, where the directivity is the same for all elements, the directivity factors out of the sum. This is known as the product theorem [16]. The normalized magnitude of the sum itself, i.e., the sum over the isotropic elements, is usually referred to as the *array factor* and its magnitude can be written in closed form as

$$P(\theta) = \frac{1}{N}\left|\sum_{n=0}^{N-1} e^{inkd\sin\theta}\right| = \left|\frac{\sin(Nk(d/2)\sin\theta)}{\sin(k(d/2)\sin\theta)}\right|. \tag{2.10}$$

This is plotted in Fig. 2.2. Here it is seen that the array has a maximum response (main lobe) at zero degrees, which, as previously mentioned, is the "broadside"

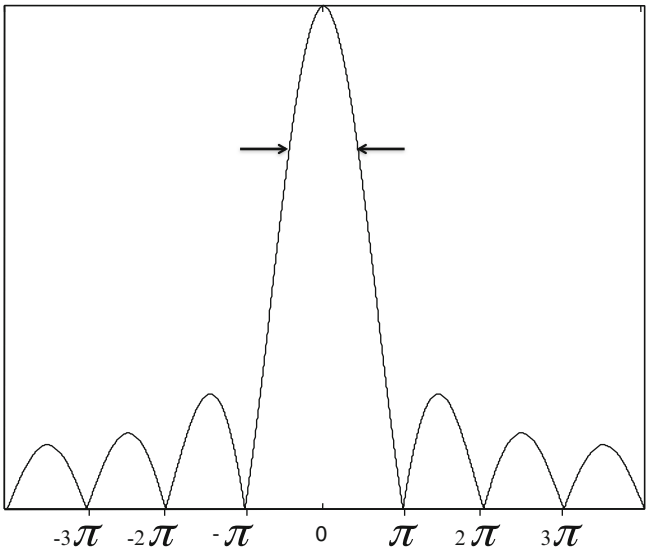

Fig. 2.2 Beam pattern of the normalized magnitude of the array factor. The *arrows* indicate the beamwidth or half-power point

direction. The other maxima are called sidelobes. This main lobe establishes a rough limit on the precision with which the bearing of an incoming plane wave can be estimated. This limit, called the beamwidth, is defined as the width of the main lobe power, $|P(\theta)|^2$, at the half-power points, or equivalently, the points where the lobe *magnitude* is $1/\sqrt{2}$ of the maximum.

It is of interest here to return to the concept of the acoustic aperture. As we have previously seen, for an array of N elements with spacing d the physical length or aperture, has length $L \approx Nd$ for N greater than 10 or so. The "acoustic" aperture, which is the aperture in units of wavelength, is called A, where $A = L/\lambda$. Then if the array has half-wavelength spacing, the acoustic aperture is, to a good approximation, equal to $N/2$. Rewriting Eq. 2.10 for the case of half-wavelength spacing, we find

$$P(\theta) = \left| \frac{\sin(\pi(N/2)\sin\theta)}{\sin((\pi/2)\sin\theta)} \right| \approx \left| \frac{\sin(\pi A\theta)}{\sin((\pi/2)\theta)} \right|, \tag{2.11}$$

where the small angle approximation for $\sin\theta \approx \theta$ has been used. Since the beam half-power points defining the beamwidth occur, to a very good approximation, when the argument of the numerator of $P(\theta)$ is $\pi/2$, i.e., when $\pi A\theta = \pi/2$, θ must be one half of the beamwidth in radians. Defining the beamwidth in radians as $\Delta\Theta$, we have a spatial "uncertainty" relation, which states that the acoustic aperture A, times the beamwidth in radians is on the order of unity. That is

$$A\Delta\Theta = (L/\lambda)\Delta\Theta \approx 1. \tag{2.12}$$

This relation offers a highly useful way to envision a line array. If one wishes to decrease the beamwidth of a line array, the acoustic aperture must be increased. For example, for a 16 wavelength array, the beamwidth can be expected to be approximately 1/16 radians. This reduces to 57.3/16 which is 3.6°. A useful rule of thumb then is that the beamwidth in degrees of a line array is about 60 divided by the aperture in wavelengths.

2.3 Beamforming

All of the above discussion centered on the "unsteered" line array. This means that the array has a maximum response to a signal arriving at broadside. If we desire a maximum response for a signal arriving at some other angle, the array must be modified or "steered" to this angle. In other words, the maximum array response must be able to accommodate signals arriving at any desired angle. The methods for doing this are discussed next.

2.3.1 Delay and Sum Beamforming

Delay and sum beamforming is accomplished by introducing time delays that set the relative phases between elements to zero when the incoming signal is at the desired angle. We begin with a weighted form of Eq. 2.2. Although the weights are not necessary for the present discussion, they will play a role in array optimization, which we will address later.

To steer the array to angle θ_m, the weighted array output expression becomes

$$S(\theta, \theta_m) = \sum_{n=0}^{N-1} w_n s_n(t - \tau_{n,m}) = \sum_{n=0}^{N-1} w_n s_n(t - n(d/c)\sin\theta_m), \tag{2.13}$$

where $\tau_{n,m} = n(d/c)\sin\theta_m$ is the time delay applied at the nth receiver to steer the array to the angle θ_m. The delay and sum procedure is depicted in Fig. 2.3. The advantage of delay and sum beamforming is that it is carried out in the time domain and therefore is valid for the broadband case, but it can engender undesirable computational problems when a digitized basebanded signal is used. This arises from the fact that a basebanded signal is inherently at a sample rate of only a few samples per cycle of the highest frequency, which means that there does not exist sufficient time scale precision when selecting a delayed version of the signal. This can be overcome by using an interpolation filter, where the signal can be interpolated and resampled [9, 10] at a higher rate just prior to the beamformer. In those cases where this is not practical, the so-called $k - \omega$ beamformer is used. The $k - \omega$ beamformer is essentially a method of using the narrowband phase shift beamformer to accomplish broadband beamforming, and is discussed in the following.

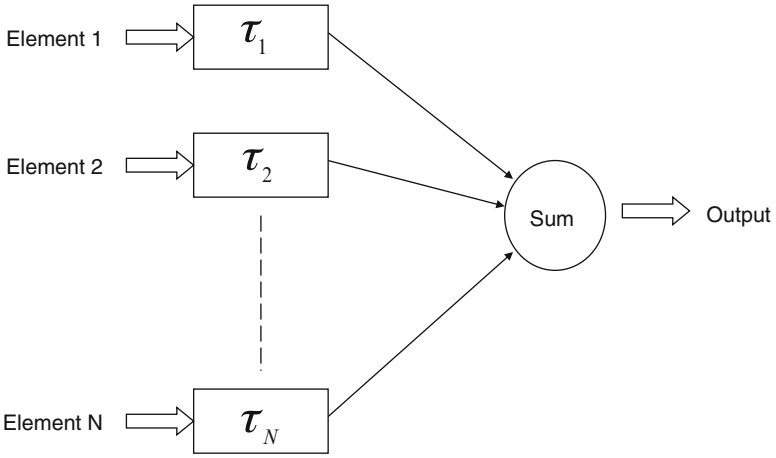

Fig. 2.3 Time domain beamformer. The outputs of the individual element time domain signals are delayed and these delayed signals are then summed, providing the output time series. This figure depicts the unweighted case

2.3.2 *Phase Shift Beamforming and the $k-\omega$ Beamformer*

The $k-\omega$ beamformer is based on the narrowband phase shift beamformer. For a given frequency, the time delays associated with a particular steering direction are associated with a phase shift, as was seen in Eq. 2.7. Introducing weights into Eq. 2.8, we have

$$S(\theta) = \sum_{n=0}^{N-1} w_n e^{inkd\sin\theta}. \tag{2.14}$$

We now introduce a steering phase shift $\phi_{n,m} = nkd\sin\theta_m$ where n and m label the element and steering angle, respectively. The steered form of Eq. 2.14 is

$$S(\theta, \theta_m) = \sum_{n=0}^{N-1} w_n e^{(inkd\sin\theta - inkd\sin\theta_m)}. \tag{2.15}$$

For computational purposes, this form can be cast into the form of a *spatial* Fourier transform [11, 18] as follows. Define the vector wavenumber $\mathbf{k} = k\sin\theta$. Equation 2.15 can now be rewritten as

$$S(\theta, \theta_m) = \sum_{n=0}^{N-1} w_n e^{ind\mathbf{k}} \times e^{-ind\mathbf{k}_m}. \tag{2.16}$$

Identifying $A_n = w_n e^{ind\mathbf{k}}$ as the signal from the nth receiver, Eq. 2.16 takes the form

$$S(\theta_m) = \sum_{n=0}^{N-1} A_n e^{-ind\mathbf{k}_m}, \tag{2.17}$$

where the dependence on θ has been suppressed. If we now make the identification

$$d\mathbf{k}_m = dk\sin\theta_m = d(2\pi/\lambda)\sin\theta_m = 2\pi(m/N). \tag{2.18}$$

Equation 2.17 takes the form of the discrete Fourier transform

$$S(\theta_m) = \sum_{n=0}^{N-1} A_n e^{-i2\pi(\frac{nm}{N})}. \tag{2.19}$$

This creates a set of N "beam bins." Note that this means that the beams are uniformly spaced in $\sin\theta$ instead of θ, so that an interpolation must be done to render the beams uniformly spaced in angle space.

If we now consider the input signal A_n itself to be the output of a *temporal* Fourier transform, then each application of Eq. 2.19 at the kth frequency becomes

$$S_k(\theta_m) = \sum_{n=0}^{N-1} A_{n,k} e^{-i2\pi(\frac{nm}{N})}, \tag{2.20}$$

where n is the receiver index and m is the beam index. This provides a narrowband beamformer for the input signal at that particular temporal frequency. Thus, a temporal Fourier transform of the time domain signal at the output of each of the N receivers is carried out. The N outputs for each index k are then put through the spatial Fourier transform, providing a set of N beams for that particular frequency.

Combining the outputs for each frequency requires a bit of effort. The problem is that the beam angles are dependent on the frequency. Following Maranda [6], we define the dimensionless wavenumber $\boldsymbol{\kappa}_m$ for the beam angle θ_m by multiplying $\mathbf{k}_m$ by d, the element spacing. Thus,

$$\boldsymbol{\kappa}_m = d\mathbf{k}_m = \pi(2d/c)f\sin\theta_m, \tag{2.21}$$

and observe that $c/2d = f_d$ is the so-called design frequency of the array,[2] i.e., the frequency at which the wavelength is twice d, the element spacing. A normalized dimensionless wavenumber can be written as

$$\boldsymbol{\kappa}_m = \pi(f/f_d)\sin\theta_m. \tag{2.22}$$

[2]The design frequency is that frequency associated with an element spacing of one half wavelength. This element spacing is sometimes referred to as spatial Nyquist sampling, since at frequencies higher than this, the beam pattern will "alias" or in the older terminology, grating lobes will be produced, due to the spatial undersampling of the signal.

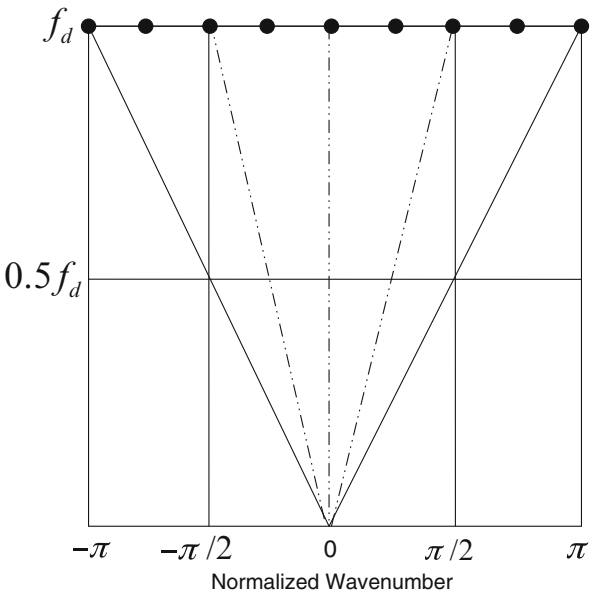

Fig. 2.4 The k-omega plot for an eight element line array. The *slanted lines* are the beams for a given value of m, the beam index. The *vertical line* is the case for $m = 0$, the broadside beam. Note that for a given beam, the wavenumber depends upon the frequency. For clarification see the discussion following Eq. 2.22

This can be plotted on a so-called $k-\omega$ plot as shown in Fig. 2.4. Here the beams are the straight line plots of frequency as a function of κ with θ as a parameter. The lines for endfire, i.e., $\kappa = \pm\pi(f/f_d)$, are for the case of $\theta = \pm 90°$. Note, however, that at $f = 0.5f_d$, that the case for $\kappa = \pi$ lies outside of the $\pm 90°$ lines. This is called the non-acoustic region and corresponds to acoustic radiation traveling at a speed less than c. Generally, for a given beam angle, an algorithm that combines the frequency components in such a way as to remain on the line of constant angle is required. For example, the two dashed lines to the left and right of the center dashed line (broadside beam) correspond to the $\pm 30°$ beams. At $f = f_d$ this is the case for $\kappa = \pm\pi/2$ but at half this frequency $\kappa = \pm\pi$, which corresponds to endfire. This must be taken into account in any broadband $k-\omega$ beamformer. The reader is directed to [6, 7] for more information on frequency domain beamforming.

2.3.3 Beam Patterns

The beam pattern of a passive array is the response of the array to a plane wave[3] acoustic signal as a function of incoming angle of the signal and, unless otherwise specified, is based on the so-called far-field approximation. This is the region where

[3] A plane wave is a wave where the surfaces of constant phase are planes and the direction of propagation is normal to these planes.

the angular field distribution is essentially independent of the distance from the array. For the case of a line array of aperture L, its far field is that distance where R satisfies the condition $R \geq L^2/(4\lambda)$.

Figure 2.2 is a beam pattern for a line array. However, this is a two-dimensional picture. The line array of isotropic elements actually has an axially symmetric three-dimensional pattern. Thus, the main beam actually has a wheel-like structure and consequently a line array can only provide directionality in a two-dimensional sense, whereas the planar array is capable of providing a true beam in three-dimensional space.

Of course, this capability in not unique to circular planar arrays, but is true of planar arrays in general. In fact, Fig. 2.2 is also the beam pattern of a rectangular array. If the rectangular planar array is in the $x-y$ plane of a Cartesian coordinate system, then Fig. 2.2 is the pattern based on the aperture along the y-axis when viewed along the x-axis. The three-dimensional beam pattern is then the product of the x-axis and y-axis beam patterns. This result is based on the product theorem discussed earlier.

2.4 Array Gain and the Directivity Index

When considering detection performance, the benefit provided by an array is the improvement in the signal-to-noise ratio. This improvement is quantified by the array gain or AG. It is defined in decibels as

$$10\log_{10}(\mathrm{SNR}_{\mathrm{out}}/\mathrm{SNR}_{\mathrm{in}}), \tag{2.23}$$

where the numerator is the signal-to-noise ratio at the array output, and the denominator is the signal-to-noise ratio at a single element of the array, assumed to be the same for all elements of the array. For a linear array two approaches exist for computing or estimating the array gain. One approach involves the directional patterns of the signal and noise fields in which the array is placed, together with the beam pattern of the array.

Let the signal and noise fields be characterized by the directional functions $S(\theta,\phi)$ and $N(\theta,\phi)$, representing the signal and noise power per unit solid angle, respectively, incident on the array from the polar directions θ and ϕ, and let $b(\theta,\phi)$ be the beam pattern of the array. Then the array gain as just defined becomes

$$\mathrm{AG} = \frac{\int_{4\pi} S(\theta,\phi)b(\theta,\phi)d\Omega \Big/ \int_{4\pi} N(\theta,\phi)b(\theta,\phi)d\Omega}{\int_{4\pi} S(\theta,\phi)d\Omega \Big/ \int_{4\pi} N(\theta,\phi)d\Omega}. \tag{2.24}$$

In this cumbersome expression each integral is merely the directional pattern of signal or noise, weighted or multiplied by the array beam pattern and integrated over the full solid angle of 4π steradians. For a single, isotropic array element, it has been assumed that $b(\theta,\phi) = 1$.

An alternative and more useful approach to array gain explicitly uses the coherence properties of the signal and noise across the dimensions of the array. In terms of the array output, the coherence is quantitatively measured by the correlation, which is a measure of the degree of similarity of the outputs between any two elements of the array. If $v_i(t)$ and $v_j(t)$ are voltages generated by two array elements, then the cross correlation coefficient between them is defined as

$$\rho_{i,j} = E\{v_i(t)v_j(t)\}. \tag{2.25}$$

The E indicates the expected value. In practice, this is usually approximated by taking the time average.[4] Now consider a line array of N elements of equal sensitivity. Let the noise-free signal amplitudes, including any phase shifts or delays incorporated for steering, be denoted by $s_i(t)$. Then the signal vector is given as

$$S = [s_1 \;\; s_2 \;\; \cdots \;\; s_N]^\dagger, \tag{2.26}$$

with † indicating the conjugate transpose. The explicit time dependence has been suppressed for simplicity. Denoting the weighting vector as W, the array output power can now be expressed as

$$P = E\{|W^\dagger S|^2\} = W^\dagger E\{SS^\dagger\}W = W^\dagger RW, \tag{2.27}$$

where R is called the covariance matrix. From the above definition of cross correlation, we see that the elements of R are $\rho_{i,j}$.

Assuming additive noise, the received signal is written as

$$v_i = s_i + n_i. \tag{2.28}$$

That is, v_i, the received signal at the ith receiver, is equal to the sum of the noise-free signal s_i and the measurement noise, n_i. Substituting this into the above expression for the covariance matrix results in

$$R = E\{(S+N)(S+N)^\dagger\}, \tag{2.29}$$

with $N = [n_1 \;\; n_2 \;\; \cdots \;\; n_N]^\dagger$. Expanding this and noting that for the case where the signal and noise are uncorrelated, the cross terms $E\{S^\dagger N\}$ are zero, the result is

$$R = E\{SS^\dagger\} + E\{NN^\dagger\} = R_s + R_n, \tag{2.30}$$

[4]The length of time one uses in computing a time average is always open to some ambiguity, since it assumes statistical stationarity and a geometrically fixed measurement scenario. Since complete information regarding these issues is never available, the shortest averaging time consistent with reasonable results should be used.

and the array output power can now be written as

$$P = W^{\dagger} R_s W + W^{\dagger} R_n W. \tag{2.31}$$

It is now possible to write the SNR out of the array as

$$\mathrm{SNR}_{\mathrm{out}} = P_s / P_n = (W^{\dagger} R_s W)/(W^{\dagger} R_n W). \tag{2.32}$$

It is of interest to look at the special case of unity weighting (all $w_i = 1$) with perfectly spatially correlated signal and spatially uncorrelated noise. Physically, this is the case of a plane wavefront across the full aperture of the array and the interelement correlations of the noise equal to zero. This means that $R^s_{i,j} = s^2$ and $R^n_{i,j} = \delta_{i,j} n^2$ with $\delta_{i,j}$ being the Kronecker delta, i.e., $\delta_{i,j} = 1$ for $i = j$ and $\delta_{i,j} = 0$ for $i \neq j$. The resulting output SNR now reduces to

$$\mathrm{SNR}_{\mathrm{out}} = (N^2 s^2)/(N n^2) = N(s^2/n^2) = N \times \mathrm{SNR}_{\mathrm{in}}, \tag{2.33}$$

so that the array gain for this case is simply N, the number of elements in the array. The array gain for the case of spatially uniform noise is called the directivity factor D. The directivity index or DI, which is the term used in the sonar equation to include the array effects, is defined as

$$\mathrm{DI} = 10 \log_{10} D. \tag{2.34}$$

Such a case of spatially uncorrelated noise occurs for a line array of equally spaced elements, with half-wavelength spacing.[5] From this expression, it would appear that one could increase the array gain by simply increasing the number of elements. However, if this is done, the noise covariance matrix will no longer be diagonal.

In the broadband case, the noise covariance is never diagonal. In this case, the array spacing is usually set to be one half wavelength at the high end of the band, i.e., the design frequency mentioned in the previous section. If this is not done, those higher frequencies where the wavelength is less than twice the element spacing, fold over, meaning aliasing will occur, just as in the case of a Fourier transform of an undersampled time-domain signal. Thus, once the design frequency is specified, the only way to increase the array gain by increasing the number of elements is to make the array physically longer, in order to preserve the desired covariance matrix structure.

[5] It is of interest to point out that the DI of a uniformly shaded line array is independent of the steering angle, in spite of the fact that steering off broadside broadens the beamwidth. This is a consequence of the fact that the beam is actually three-dimensional. As the steering goes away from broadside, the beam takes on a conical shape with a decreasing cone angle.

Because of propagation effects, it can be possible that the signal itself is not totally coherent along the full aperture of a towed array. In this case, the structure of R_s in Eq. 2.32 degrades so that full advantage of the aperture cannot be exploited. In this case, even if we were to lengthen the array while maintaining the design frequency requirement, the performance would degrade. In other words, for a spatially uniform noise field, spacing the elements of a line array at half wavelength is a necessary condition for achieving an array gain of $10\log_{10}N$, but not sufficient. It is also necessary to have a signal that is perfectly coherent across the aperture. That is, the signal coherence length must be at least equal to the aperture length.

2.4.1 Limitations of the Directivity Index

The DI is still a useful quantity for many applications. For simple arrays, such as the line and circular plane, the DI can be evaluated in closed form. For arrays that cannot be approximated by these simple forms, the DI can be found numerically if the beam pattern is known.

Because it can be easily evaluated for some common array configurations, the directivity index is a useful parameter for providing at least a first-cut estimate of the gain of an array. Yet its restriction to the special case of a signal with perfect spatial coherence in isotropic noise must not be overlooked. In the real ocean these ideal conditions seldom, if ever, occur. For example, the noise background of the sea is known to be anisotropic, and to have directionality in both the vertical and (at low frequencies) the horizontal planes. Transmitted signals commonly are received from a number of different vertical directions over a variety of refracted and reflected multipaths. Such multipath propagation causes the spatial coherence of a low-frequency signal to decrease rapidly with range and hydrophone separation [15], so that, when an array is steered toward the signal arriving along one of the multipaths, the signals arriving from the others act as noise and produce a degraded array gain. Studies by Carey [3] at frequencies near 400 Hz show for the deep-water cases that coherence lengths on the order of 100 wavelengths can be achieved to ranges of 500 km; while in the variable downward refraction conditions of shallow-water waveguides with sand-silt bottoms, coherence lengths are on the order of 30 wavelengths out to ranges of 45 km. Because of this, directivity index, although it is still a useful sonar parameter for approximate calculations, should be used with caution in the complicated signal and noise environment of the real ocean. Whenever the spatial coherence characteristics of signal and noise are known, or can be estimated, array gain should replace DI in realistic sonar calculations.

2.5 Array Optimization

There are different approaches to optimizing an array. Probably the simplest approach is to try to obtain sidelobes as low as possible, given a desired beamwidth [1, 14]. The set of coefficients $\{w_i\}$ consistent with this goal usually results in a set of "tapered" coefficients, i.e., large in the center of the aperture and monotonically decreasing toward the ends of the aperture. Generally speaking, the lower the sidelobes, the wider the beamwidth. This is consistent with the beamwidth/aperture product discussed previously, since tapered shading reduces the effective length of the aperture.

Another approach seeks the maximum SNR for a given spatial noise distribution. That is, an attempt is made to directly account for the character of the noise field. This entails expressing the array gain as a function of the shading coefficients $\{w_i\}$, and finding the set of coefficients that maximize it. That is, we maximize

$$\mathrm{AG} = (W^{\dagger} R_s W)/(W^{\dagger} R_n W). \tag{2.35}$$

Although for the isotropic noise case, the array gain can never exceed N, there are cases when the character of the noise field permits array gains greatly exceeding N. Consider the example of a noise field with a spatial distribution where the noise level is high in signal-free regions and low near the angle of arrival of the signal. Clearly, any set of coefficients that desensitizes the array in the direction of the noise will allow performance that exceeds that for a spatially uniform noise field. Thus, if the noise covariance is known, then a set of weights can be computed for the maximum *SNR*. In the case where the array must deal with multiple sources, however, the *spatial resolution* of the arriving signals angles becomes of prime importance. Improving resolution is not necessarily consistent with maximizing the gain. An example of a popular adaptive processor providing high spatial resolution is the so-called *MVDR* or minimum variance distortionless response array [2]. This technique minimizes the array output power over the full solid angle while constraining the response in the direction of interest to be fixed at unity. The term "minimum variance," therefore, does not refer to the variance on the bearing estimate, but on the minimum noise power (i.e., noise variance). The *MVDR* weights for an array with noise covariance R_n is given by [5]

$$W_{\mathrm{mvdr}} = \frac{R_n^{-1} S}{S^{\dagger} R_n^{-1} S} \tag{2.36}$$

with S being the signal vector. This set of weights provides an array with a narrow main lobe. As mentioned above, high-resolution beamformers are generally not optimal in a SNR sense, but can provide the ability to resolve sources that are closely spaced in angle, where the conventional beamformer would fail. A complete discussion of these types of beamformers is found in [8, 17].

Note that if one has knowledge of the noise covariance as a function of time, Eq. 2.36 provides a means of adapting the weights to maintain performance. Because of this, the MVDR is sometimes referred to as an adaptive beamformer. It is not unique in this sense, however, since any covariance-based algorithm that allows adaptation of the weights leads to an adaptive array. The field of adaptive arrays is a large one and the reader is directed to [17] for further information.

2.6 Bearing Estimation

In the passive sonar problem, the estimation of the bearing of a source is a task often required of a towed line array. Assuming that range of the source is large as compared to the aperture of the array, it can be assumed that the arriving signal is a plane wave. Also, to a very good approximation, this can be considered to be a two-dimensional problem, as depicted in Fig. 2.1. The problem then is to estimate the angle θ. In practice, this is done by seeking the maximum of the beamformer output. From an heuristic point of view, this seems like the obvious thing to do. From an estimation theory viewpoint, it is of interest to ask if this approach is optimum. The answer is yes and is shown for the narrowband case by the following. In the case of spectral estimation the maximum likelihood estimate of the frequency is found from the peak of the periodogram $P(f)$, where the periodogram is defined as the normalized magnitude squared of the frequency spectrum [5], which is given by,

$$P(f) = |(S(f)|^2. \tag{2.37}$$

Since we have shown that the beam pattern of an array is the Fourier transform of the aperture, it is a spectrum in $\mathbf{k}$ space. Then the periodogram is

$$P(\mathbf{k}) = \left| \sum_{n=0}^{N-1} a_n e^{in\mathbf{k}d} \right|^2, \tag{2.38}$$

with

$$\mathbf{k} = kd\sin\theta. \tag{2.39}$$

Since the bearing is related to the frequency through the phase term, i.e.,

$$kd\sin\theta = 2\pi(f/c)d\sin\theta, \tag{2.40}$$

we can use the maximum likelihood invariance[6] theorem [19] to conclude that the maximum of $P(\mathbf{k})$ is the ML estimator for the bearing angle θ.

[6]This theorem [19] states that given an invertible relation between two variables, say $y = f(x)$, the maximum likelihood estimate of x, designated as $\hat{x}$, can be found by finding the maximum likelihood estimate of y, designated as $\hat{y}$, and solving the relationship for $\hat{x}$. That is, $\hat{y} = f(\hat{x})$.

From an estimation theory point of view, the quality of an estimate is measured by the error variance [5] on the estimate, where this variance is given by

$$\sigma_\theta^2 = E\{[\theta - \hat{\theta}]^2\}. \tag{2.41}$$

The Cramér–Rao lower bound or *CRLB* for an estimate is the smallest variance on an estimate that can be achieved by an estimator, and an estimator that does achieve it is called *efficient* [5]. For the case of narrowband bearing estimation using a line array, for a singe time sample[7]

$$\sigma_\theta^2 = \frac{3}{(\text{SNR})\pi^2 N \frac{N+1}{N-1}(A\text{cos}\theta)^2}, \tag{2.42}$$

which for N reasonably large, is well approximated by

$$\sigma_\theta^2 = \frac{3}{(\text{SNR})\pi^2 N(A\text{cos}\theta)^2}. \tag{2.43}$$

Note that the acoustic aperture $A = L/\lambda$ appears in the denominator. As expected, the CRLB on the estimate of θ decreases with an increase in A. Also, it decreases with an increase in the *SNR*, and N, the number of elements in the array. Recall also that the beamwidth of the array is well approximated by the reciprocal of the acoustic aperture, so we see that the variance on the estimate of the bearing decreases with the beamwidth, as expected. The term $A\text{cos}\theta$ is usually referred to as the *projected aperture*. Thus, the CRLB on the estimate of θ *increases* as the bearing moves away from broadside, since the projected aperture *decreases*.

2.7 Three-Dimensional Arrays

Although the line array plays a major role in underwater signal processing, three-dimensional arrays can arise in many applications. For example, arrays may be mounted on the hulls of autonomous undersea vehicles (AUVs), leading to conformal arrays. Submarines and surface ships carry spherical and cylindrical arrays. Although the mathematical modeling of such arrays easily obtains from the generalization of the previous discussion of the line array, the optimization of such arrays becomes much more problematical, and usually must be treated numerically.

The general expression for a three-dimensional array, in a standard polar coordinate system, is given by

[7]For our purposes here, the single time sample case is sufficient. The more general case, which includes the number of independent time samples, will be discussed in Chap. 5.

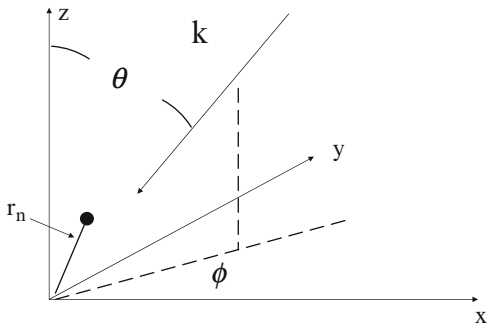

Fig. 2.5 Coordinate system for the three-dimensional array. The reference phase plane passes through the origin and the incoming wave is normal to this plane

$$S(\theta, \phi) = \sum_{n=1}^{N} w_n e^{i\mathbf{k}\cdot\mathbf{r_n}}. \tag{2.44}$$

Referring to Fig. 2.5, $\mathbf{k}$ is the wave vector of the incoming plane wave signal. Its direction defines the direction of propagation and has the magnitude $|\mathbf{k}| = k = 2\pi/\lambda$. θ and ϕ are the polar angles of $\mathbf{k}$ and $\mathbf{r_n}$ is the radial coordinate of the nth isotropic receiver element. To see the structure of the phase term in Cartesian coordinates, it is useful to expand it as follows.

$$\mathbf{k}\cdot\mathbf{r_n} = k[x_n \sin\theta\cos\phi + y_n \sin\theta\sin\phi + z_n\cos\theta]. \tag{2.45}$$

For a line array on the x-axis, y_n, z_n, and ϕ are all zero and Eq. 2.45 reduces to

$$\mathbf{k}\cdot\mathbf{r_n} = knd\sin\theta, \tag{2.46}$$

which is the phase term for a line array for $x_n = nd$. Note that an element directivity can be included here, just as in the line array case, but the product theorem no longer holds, since these directivities will not all be in the same rotational positions, except for the case of a planar array, where all the z_n are zero.

One case of interest is the *conformal* array, i.e., an array that conforms to a surface. By defining the *projected planar array* (PPA) [4] to be the array that obtains by placing one axis of the coordinate system, say the z axis, along the maximum of the main lobe, and setting the z coordinates of all the elements to zero, some insight into the performance of the associated conformal array can be obtained. First, it can be shown [13] that the main lobe of both the conformal array and its associated PPA are essentially the same. Second, the apertures of the PPA are inversely related to the beamwidths in the $x-z$ and $y-z$ planes of the PPA. However, the sidelobe behavior of the conformal array is generally worse than that of the associated PPA. By worse, we mean that, for a given set of shading coefficients, the sidelobes of the conformal array are higher than those of the PPA, and the difference increases as the observation angle moves away from the MRA.

Examples of beamforming for three-dimensional arrays are given in [4, 12, 13].

References

1. Burdic, W.S.: Underwater Acoustic Systems Analysis. Prentice Hall, Englewood Cliffs (1991)
2. Capon, J., Greenfield, J., Kolker, R.J.: Multidimensional maximum-likelihood processing of a large aperture seismic array. Proc. IEEE **55**, 192–211 (1967)
3. Carey, W.M.: The determination of signal coherence length based on signal coherence and gain measurements in deep and shallow water. J. Acoust. Soc. Am. **104**, 831 (1998)
4. Kelly, J., Sullivan, E., Salisbury, J.: Comments on "conformal array beam patterns and directivity indices". J. Acoust. Soc. Am. **65**(2) (1979) [J. Acoust. Soc. Am. **63**, 841–847 (1978)]
5. Kay, S.M.: Fundamentals of Statistical Signal Processing; Estimation Theory. Addison-Wesley, Reading (1991)
6. Maranda, B.: Efficient digital beamforming in the frequency domain. J. Acoust. Soc. Am. **86**, 1813–1819 (1989)
7. Mucci, R.A.: A comparison of efficient beamforming algorithms. IEEE Trans. ASSP. **ASSP-32**, 548–558 (1984)
8. Monzingo, R.A., Miller, T.W.: Introduction to Adaptive Arrays. Wiley, New York (1980)
9. Pridham, R.G., Mucci, R.A.: A novel approach to digital beamforming. J. Acoust. Soc. Am. **63**, 425–434 (1978)
10. Pridham, R.G., Mucci, R.A.: Digital interpolation beamforming for low-pass and bandpass signals. Proc. IEEE **67**, 904–919 (1979)
11. Rudnick, P.: Digital beamforming in the frequency domain. J. Acoust. Soc. Am. **46** (1968)
12. Sullivan, E.J.: Amplitude shading of irregular acoustic arrays. J. Acoust. Soc. Am. **63** (1978)
13. Sullivan, E.J.: The sidelobe behavior of conformal arrays. J. Acoust. Soc. Am. **71** (1982)
14. Taylor, T.T.: Design of line source antennas for narrow beamwidth and low side lobes. IRE Trans. **AP-3** (1955)
15. Urick, R.J.: Measurements of the vertical coherence of the sound from a near-surface source in the sea and the effect on the gain of an additive vertical array. J. Acoust. Soc. Am. **54** (1973)
16. Urick, R.J.: Principles of Underwater Sound in the Sea. McGraw-Hill, New York (1975)
17. van Trees, H.L.: Optimum Array Processing. Wiley, New York (2002)
18. Williams, J.R.: Fast beamforming algorithm. J. Acoust. Soc. Am. **44** (1968)
19. Zehna, P.W.: Invariance of maximum likelihood estimators. Ann. Math. Stat. **37** (1966)

Chapter 3
Statistical Signal Processing Overview

3.1 Introduction

Before proceeding to a discussion of the Kalman filter and it variants, it is helpful to present an overview of statistical signal processing. Although the concern of this book is mainly estimation, this chapter begins with a short discussion of detection theory. There are two reasons for this. One is for completeness and the other is that there will be some mention of detection in Chap. 6 when the innovations sequence, one of the major components of the Kalman filter, is discussed. Consequently, the treatment here is not to be considered as a comprehensive discussion of detection theory.

The section on estimation theory will begin with a discussion of the Cramér–Rao lower bound (CRLB), which is a lower bound on the variance achievable by a given estimator. And since we are dealing with improvements in estimation via the use of models, a quantitative measure of the improvement is required. Since the variance is the accepted measure of the quality of an estimate, the CRLB is an essential tool for this measure.

3.2 Detection Theory for Totally Known Signals

In the field of underwater signal processing, the binary hypothesis describes most detection problems. That is, the concern is with determining whether a signal does or does not exist. Such problems are most often dealt with by the Neyman–Pearson (N–P) detector [7, 11]. The N–P detector delivers the highest probability of detection for a given probability of false alarm. It allows for two kinds of errors. Deciding that the signal is present when it is not is called an error of Type I, and deciding that the signal is *not* present when it is, is an error of Type II.

E.J. Sullivan, *Model-Based Processing for Underwater Acoustic Arrays*,
SpringerBriefs in Physics, DOI 10.1007/978-3-319-17557-7_3

As a simple example of the N–P detector, consider the problem of detecting the presence of a nonzero DC level in white Gaussian noise. The binary hypothesis is stated as follows. Designating the DC level as v, the zero-mean white Gaussian noise as n with variance σ^2 and the received signal as y, two hypotheses are considered. These are H_0, the hypothesis that no signal is present, and H_1, the hypothesis that a signal is present. These hypotheses, for the case of real signals, are stated as follows.

$$\begin{aligned} H_0 &: \ y = n \\ H_1 &: \ y = v + n. \end{aligned} \tag{3.1}$$

A probabilistic description of these hypotheses, based on the assumption of n being white and Gaussian, is given by

$$p(y|H_0) = \frac{1}{(2\pi\sigma^2)^{1/2}} e^{-y^2/2\sigma^2}, \tag{3.2}$$

and

$$p(y|H_1) = \frac{1}{(2\pi\sigma^2)^{1/2}} e^{-(y-v)^2/2\sigma^2}. \tag{3.3}$$

The N–P criterion states that the optimal detection structure is given by the so-called *Likelihood Ratio*, which is defined by

$$L = \frac{p(y|H_1)}{p(y|H_0)} = \frac{e^{-(y-v)^2/2\sigma^2}}{e^{-y^2/2\sigma^2}}. \tag{3.4}$$

The N–P detector decides that H_1 is true if $L > \gamma$ where as will be seen, γ is determined by the desired probability of false alarm, and that H_0 is true otherwise. In conventional practice, the logarithm of L is used so that the test statistic, lnL, is compared with $ln\gamma$. From Eq. 3.4, the logarithm is compared with $ln\gamma$. The logarithm of the likelihood ratio is

$$lnL = -(y-v)^2/2\sigma^2 + y^2/2\sigma^2 = [2vy - v^2]/2\sigma^2, \tag{3.5}$$

so that H_1 is true if

$$\frac{yv}{\sigma^2} > \frac{v^2}{2\sigma^2} + ln\gamma. \tag{3.6}$$

This is cast in terms of probability as follows. The probability of detection, i.e., the probability that H_1 is true, is given by

$$P_D = \frac{1}{(2\pi\sigma^2)^{1/2}} \int_T^\infty e^{-(y-v)^2/2\sigma^2} dy, \tag{3.7}$$

and the probability of false alarm, P_{FA}, is given by

$$P_{\text{FA}} = \frac{1}{(2\pi\sigma^2)^{1/2}} \int_T^\infty e^{-y^2/2\sigma^2} dy, \tag{3.8}$$

and T is the threshold. The threshold is the level above which a detection is declared. The procedure for implementation of the N–P detector now follows immediately. The desired P_{FA} is chosen, which defines the threshold T via Eq. 3.8. Given T, Eq. 3.7 is then used to compute P_D. The detection problem then is not completely defined without the specification of P_{FA}.

The above example is based on a single sample. In the case of a known, discretely sampled time-varying signal v_n with N samples, the above argument is generalized as follows. Still assuming zero-mean white (independent) Gaussian noise and a totally known signal with N samples, Eq. 3.2 generalizes to the product of the individual likelihoods.[1] Equation 3.2 is replaced with

$$p(\mathbf{y}|H_0) = \frac{1}{(2\pi\sigma^2)^{N/2}} e^{-\frac{1}{2\sigma^2}\sum_{n=1}^N y_n^2}, \tag{3.9}$$

and Eq. 3.3 is replaced with

$$p(\mathbf{y}|H_1) = \frac{1}{(2\pi\sigma^2)^{N/2}} e^{-\frac{1}{2\sigma^2}\sum_{n=1}^N (y_n - v_n)^2}. \tag{3.10}$$

Here, $\mathbf{y}$ represents the vector of N data samples with elements $\{y_n\}$. For this case Eq. 3.6 generalizes to

$$\frac{1}{\sigma^2}\sum_{n=1}^N y_n v_n > \frac{1}{2\sigma^2}\sum_{n=1}^N v_n^2 + ln\gamma. \tag{3.11}$$

We see that, as one would expect, the threshold increases with the signal energy, which is given by

$$E = \sum_{n=1}^N v_n^2. \tag{3.12}$$

Upon rearranging Eq. 3.11, we finally get

$$\sum_{n=1}^N y_n v_n > \frac{1}{2}\sum_{n=1}^N v_n^2 + \sigma^2 ln\gamma. \tag{3.13}$$

[1]This is clear if the signal is sampled at the so-called Nyquist rate, since these probabilities are independent. When the sample rate is higher than the Nyquist rate, no new independent samples are being introduced so that this likelihood function still holds.

Note that the term on the LHS of Eq. 3.13 is the correlation between the data and the signal. Thus, when the form of the signal is known a priori, this test statistic can be computed. This is the well-known matched filter, and in the form shown, is known as the *Replica Correlator* [2]. The recipe for its implementation now follows from the generalization of Eqs. 3.7 and 3.8 which is found by simply replacing the likelihoods for the single sample case with those for the N sample case. Thus, the probability of detection now becomes

$$P_D = \frac{1}{(2\pi\sigma^2)^{N/2}} \int_T^\infty e^{-\frac{1}{2\sigma^2}\sum_{n=1}^N (y_n - v_n)^2} d\mathbf{y}, \tag{3.14}$$

and the probability of false alarm becomes

$$P_{\text{FA}} = \frac{1}{(2\pi\sigma^2)^{N/2}} \int_T^\infty e^{-\frac{1}{2\sigma^2}\sum_{n=1}^N y_n^2} d\mathbf{y}. \tag{3.15}$$

Selecting a value for the probability of false alarm (Eq. 3.15) is solved for the lower limit on the integral, which yields the threshold value. Given this threshold, one can then directly compute the detection probability from Eq. 3.14. This provides the basis for a family of equations for the detection probability as a function of the false alarm probability with the signal-to-noise ratio as a parameter specifying each curve in the family. Such a family of curves is called the receiver operating characteristic or ROC curves. For more information the reader is directed to [7].

For the purposes of this book, the logarithm of the likelihood for H_1 which is obtained from Eq. 3.10 will be our main interest, since it is intimately related to the innovations sequence provided by the Kalman filter. As will be seen in Chap. 4 the recursive update procedure of the Kalman filter provides a sequence called the innovations sequence, each term of which is the difference between the new measurement and the previous estimate of the measurement. Its square is then the log likelihood for white noise for a properly tuned Kalman filter. Thus a sequential log likelihood term is available is a natural product of the Kalman filter.

3.3 Classical Estimation Theory

As with the discussion of detection theory in Sect. 3.1, this discussion of estimation theory will be minimal, since it is mainly intended to provide a basis for the following chapter on the Kalman filter. Here, the term classical refers to estimators for deterministic parameters.

As mentioned in Chap. 1, estimation is the determination of the values of certain parameters of the signal, the source or the medium. It is convenient to classify the estimation task into two types: parametric and nonparametric. Parametric estimation can be further separated by identifying those cases where the parameters have a clear physical basis, such as ocean bottom density and sound speed, and those cases where

this is not necessarily so, which can be the case in time series analysis. To put in a different way, parametric estimation consists of providing some kind of parameter-based structure to the signal. This is done for either of two reasons: to improve the performance of the estimator or to extract estimates of certain parameters of the source or medium.

3.4 The Cramér–Rao Lower Bound

Before continuing, it is useful at this point to introduce the CRLB, which is the lower bound on the error variance of an estimate, since as already mentioned, the quality of an estimate is most commonly measured in terms of its variance. We begin with the problem of finding the estimate of a parameter x where the data are described by a linear model with zero mean white (uncorrelated) Gaussian noise. That is

$$y_i = x + n_i, \tag{3.16}$$

and the likelihood function is given by

$$p(\mathbf{y}|x) = \frac{1}{(2\pi\sigma^2)^{N/2}} e^{\frac{-1}{2\sigma^2}\sum_{i=1}^{N}(y_i - x)^2}, \tag{3.17}$$

with $\mathbf{y} = [y_1\ y_2\ \cdots\ y_N]$.
It is now assumed that the best estimate is the mean of the data, viz.

$$\hat{x} = \frac{1}{N}\sum_{i=1}^{N} y_i = \frac{1}{N}\sum_{i=1}^{N}(x + n_i) = E\{x\} + \frac{1}{N}\sum_{i=1}^{N} n_i. \tag{3.18}$$

Here the notation E indicates the expected value. The variance on the estimate follows as

$$\sigma_x^2 = E\{(\hat{x} - E\{x\})^2\} = E\left\{\left(\frac{1}{N}\sum_{i=1}^{N} n_i\right)^2\right\} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} E\{n_i n_j\}, \tag{3.19}$$

and since we have assumed Gaussian white (uncorrelated) noise, this reduces to

$$\sigma_x^2 = \frac{1}{N^2}\sum_{i=1}^{N} E\{n_i^2\} = \frac{\sigma^2}{N}, \tag{3.20}$$

where $\sigma^2 = E\{n_i^2\}$.[2]

[2]This is a well-known result—that the variance on the mean of a data set of length N is the variance on the data set itself divided by N.

Since the likelihood function for the data is known, it is possible to establish a lower bound on the variance of an estimate based on this likelihood. This is the CRLB [6]. An estimator that achieves this bound[3] is said to be *efficient*. It is derived here following the treatment in [5] for the case of a single real parameter x.

The expected value of the error of an unbiased estimate of x is

$$E\{\hat{x} - x\} = \int (\hat{x} - x) p(y|x) dy = 0. \tag{3.21}$$

The derivative of this with respect to x is

$$\int (\hat{x} - x) \frac{\partial p(y|x)}{\partial x} dy - \int p(y|x) dy = 0. \tag{3.22}$$

Using

$$\frac{\partial p(y|x)}{\partial x} = p(y|x) \frac{\partial lnp(y|x)}{\partial x}, \tag{3.23}$$

and the fact that p is normalized, Eq. 3.22 can be written as

$$\int (\hat{x} - x) \frac{\partial lnp(y|x)}{\partial x} p(y|x) dy = 1. \tag{3.24}$$

The integrand is now factored resulting in

$$\int (\hat{x} - x) \sqrt{p(y|x)} \left[\frac{\partial lnp(y|x)}{\partial x} \sqrt{p(y|x)} \right] dy = 1. \tag{3.25}$$

Using the Cauchy Schwarz inequality[4] on Eq. 3.25 results in

$$\left[\int p(y|x) \left(\frac{\partial lnp(y|x)}{x} \right)^2 dy \right] \left[\int p(y|x) (\hat{x} - x)^2 dy \right] \geq 1. \tag{3.26}$$

The left bracketed term in Eq. 3.26 is the Fisher information matrix (FIM)[5] I, and the right bracketed term is the error variance on the estimate of x. This states that

$$\sigma_x^2 \geq I^{-1}. \tag{3.27}$$

[3] It is important to keep in mind that the CRLB is only meaningful in terms of the associated likelihood function.

[4] This inequality states that $\left[\int A(z) B(z) dz \right]^2 \leq \int [A(z)]^2 \int [B(z)]^2 dx$.

[5] This is a one-dimensional problem so that here the Fisher matrix is one dimensional. However, the term matrix is used to avoid confusion and maintain a consistent terminology.

This states that the error variance on the estimate of x can never be less than the inverse of the FIM I. A more general derivation, dealing with multiple parameters, can be found in [7],

The likelihood function for a signal $\mathbf{y}$, that depends upon an M-dimensional parameter vector $\mathbf{x} = [x_1, x_2, \ldots, x_M]$, is the probability of $\mathbf{y}$ conditioned on $\mathbf{x}$. It is written as $p(\mathbf{y}/\mathbf{x})$, with $\mathbf{x}$ considered to be fixed. The FIM for this case is

$$I_{ij} = -E\left\{\frac{\partial}{\partial x_i}\frac{\partial}{\partial x_j} ln[p(\mathbf{y}|\mathbf{x})]\right\}. \tag{3.28}$$

The CRLB on x_k, the estimate of the kth element of $\mathbf{x}$ then follows as

$$\sigma_{x_k}^2 \geq (I^{-1})_{kk}. \tag{3.29}$$

Since n_i in Eq. 3.18 represents a white Gaussian process, it is seen from Eqs. 3.17 and 3.28 that the Fisher matrix is

$$I = \frac{N}{\sigma_x^2}, \tag{3.30}$$

which leads to the result that the lower bound on the variance of the estimate $\hat{x}$ is simply the noise variance divided by N, i.e.,

$$\text{CRLB}(x) = \frac{\sigma_x^2}{N}. \tag{3.31}$$

But this is the same as found in Eq. 3.20. Thus, the estimator of Eq. 3.20 is efficient.

3.5 Estimator Structure

The estimator used in the example above is linear. The general linear model is given by

$$\mathbf{y} = A\mathbf{x} + \mathbf{n}. \tag{3.32}$$

In the above, A is an $N \times M$ matrix and we shall refer to it here as the measurement matrix. N is the number of measurements and M is the size of the parameter vector $\mathbf{x}$. This measurement model will be used throughout this chapter.

3.5.1 The Minimum Variance Unbiased Estimator

In the case where $p(\mathbf{y}|\mathbf{x})$ is known, it can be shown from the CRLB theorem [6] that $\alpha(\mathbf{x})$ is an efficient minimum variance estimator if and only if

$$\frac{\partial lnp(\mathbf{y}|\mathbf{x})}{\partial \mathbf{x}} = \mathbf{I}(\mathbf{x})(\alpha(\mathbf{x}) - \mathbf{x}). \tag{3.33}$$

Since this is an efficient estimator, it also follows that the covariance matrix for the estimate is

$$\mathbf{C}_{\mathbf{x}} = \mathbf{I}^{-1}. \tag{3.34}$$

Note that when the expected value of Eq. 3.33 is zero, the estimator α is unbiased. As an example, consider the following case.

The Gaussian likelihood function for the linear model with white noise is given by

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{(2\pi\sigma^2)^{N/2}} e^{-\frac{1}{2\sigma^2}(\mathbf{y}-A\mathbf{x})^T(\mathbf{y}-A\mathbf{x})}. \tag{3.35}$$

From this equation, the Gaussian log likelihood (ignoring the multiplying factor which does not depend on the data) is $-\frac{1}{2\sigma^2}[(\mathbf{y}-A\mathbf{x})^T(\mathbf{y}-A\mathbf{x})]$. Then with the help of the vector calculus chain rule, $\nabla(a^Tb) = (\nabla a^T)b + (\nabla b^T)a$, with ∇ denoting the derivative with respect to the vector $\mathbf{x}$, we find that Eq. 3.33 becomes,

$$\frac{\partial lnp(\mathbf{y}|\mathbf{x})}{\partial \mathbf{x}} = \frac{1}{\sigma^2}[A^T\mathbf{y} - (A^TA)\mathbf{x}], \tag{3.36}$$

which can be rewritten as

$$\frac{\partial lnp(\mathbf{y}|\mathbf{x})}{\partial \mathbf{x}} = \frac{(A^TA)}{\sigma^2}[(A^TA)^{-1}A^T\mathbf{y} - \mathbf{x}]. \tag{3.37}$$

Upon comparison with Eq. 3.33 we find that the MVU estimator of $\mathbf{x}$ is

$$\hat{\mathbf{x}} = (A^TA)^{-1}A^T\mathbf{y}, \tag{3.38}$$

and since it is an efficient estimator, the covariance on the estimate of $\mathbf{x}$ is

$$C_x = \sigma^2(A^TA)^{-1}. \tag{3.39}$$

Although it will not be a problem in this book, in the interest of completeness, it should be pointed out that the MVU is a special case of the minimum mean square estimator (MMSE) for the case where zero bias is assumed. For the general MMSE,

an unbiased estimator cannot always be found. This is seen as follows. The general minimum mean square error is written as

$$\mathrm{MSE}(\hat{x}) = E[(\hat{x} - x)^2] = E\left\{\left[(\hat{x} - E(\hat{x})) + (E(\hat{x}) - x)\right]^2\right\}. \tag{3.40}$$

Here it is seen that the left-hand term in the RHS of this equation delivers the variance on the estimate, the right-hand term is a function of x, i.e., the estimator is biased. By assuming that the $[E(\hat{x}) - x]$ term is zero and minimizing only the variance delivers the unbiased minimum variance estimator or MVU.

3.5.2 *The Non-white Minimum Variance Unbiased Estimator*

For the case where the noise statistics are still zero mean but not white, the results for the white noise case for the Minimum Variance Unbiased Estimator can be easily generalized by using a prewhitening transformation. Suppose the colored covariance matrix R is known and factorable as $R^{-1} = D^T D$. This whitens the noise since $R = (D^T D)^{-1} = D^{-1}(D^T)^{-1}$ so that

$$DRD^T = D(D^T D)^{-1} D^T = DD^{-1}(D^T)^{-1} D^T = I, \tag{3.41}$$

and the linear model transforms as

$$D\mathbf{y} = DA\mathbf{x} + D\mathbf{n}. \tag{3.42}$$

The MVU now becomes

$$\hat{\mathbf{x}} = [(DA)^T DA]^{-1} (DA)^T D\mathbf{y}, \tag{3.43}$$

which can be rewritten as

$$\hat{\mathbf{x}} = [A^T (D^T D) A]^{-1} A^T D^T D\mathbf{y}, \tag{3.44}$$

or

$$\hat{\mathbf{x}} = (A^T R^{-1} A)^{-1} A^T R^{-1} \mathbf{y}, \tag{3.45}$$

and the covariance on $\hat{\mathbf{x}}$ becomes $C_{\hat{\mathbf{x}}} = (A^T R^{-1} A)^{-1}$.

3.5.3 Best Linear Unbiased Estimator

In the case where the Minimum Variance Unbiased Estimator cannot be found or the pdf of the data is unknown, but the covariance of the data is *known* and the expected value of $\mathbf{n}$ is zero, the approach is to find the best linear estimator that is unbiased. The minimum variance approach in this case delivers the best linear unbiased estimator (BLUE).

The BLUE is found from the classical form of the Gauss–Markov [8] theorem which is stated as follows. If the data model is linear as in Eq. 3.32, which is rewritten here

$$\mathbf{y} = A\mathbf{x} + \mathbf{n} \tag{3.46}$$

and $\mathbf{x}$ is a nonrandom vector, the covariance of $\mathbf{n}$ is R and $E\{\mathbf{n}\} = 0$, then the BLUE for $\mathbf{x}$ is

$$\hat{\mathbf{x}} = (A^T R^{-1} A)^{-1} A^T R^{-1} \mathbf{y}, \tag{3.47}$$

and the variance of the estimate is $C_{\hat{\mathbf{x}}} = (A^T R^{-1} A)^{-1}$. This is proved in [8]. The important point here is that the form is the same as for the MVU estimator, but they are not the same in terms of optimality. That is, Eq. 3.47 is optimal only under the assumption of the data being described by a linear model and only the mean and covariance of the noise are known, whereas if the pdf were known and Gaussian, then the MVU is optimal.

It now immediately follows that if $R = \sigma^2 \mathbf{I}$ and the expected value of $\mathbf{n}$ is zero, then Eq. 3.47 reduces to

$$\hat{\mathbf{x}} = (A^T A)^{-1} A^T \mathbf{y} = P\mathbf{y}, \tag{3.48}$$

which is the least squares estimate. This is the case where the knowledge of the statistics is minimal. It is still a form of BLUE however. The form of the linear estimator in Eq. 3.48, denoted by P, is sometimes referred to as the Moore–Penrose inverse [9, 10] or the pseudoinverse, since it is the solution to a least squares estimate for the overdetermined problem.

Equation 3.48 can be viewed from two different points of view. If there is only one realization of the data vector, then it is the least-squares solution for $\mathbf{x}$. On the other hand, if a sequence of data vectors is assumed, Eq. 3.48 is written as

$$\hat{\mathbf{x}}_i = (A^T A)^{-1} A^T \mathbf{y}_i = P\mathbf{y}_i, \tag{3.49}$$

and can be considered to be a sequential processor which produces a sequence of estimates of the parameter $\mathbf{x}$, i.e., $\mathbf{x}_i$. In the stationary case, this is simply an extension to two dimensions of the least-squares problem. In the nonstationary case, however, the form of Eq. 3.49 leads to a sequential estimator which is a natural lead-in to the Kalman filter, which will be discussed in the next chapter.

Another approach to the least squares estimation problem is the weighted least squares (WLS) problem. Since the least squares estimator can be found by minimizing J where

$$J = (\mathbf{y} - A\mathbf{x})^T(\mathbf{y} - A\mathbf{x}), \tag{3.50}$$

an extension can be found by including a weighting vector W, such that J becomes

$$J = (\mathbf{y} - A\mathbf{x})^T W(\mathbf{y} - A\mathbf{x}). \tag{3.51}$$

The solution can be found from minimizing J using the vector calculus chain rule. Not surprisingly the estimator for this case is

$$\hat{\mathbf{x}} = (A^T WA)^{-1}A^T W\mathbf{y}. \tag{3.52}$$

Clearly, this is not a generally optimum approach. The selection of a weighting matrix is usually dictated by the user and could be based on many different criteria.

3.5.4 The Maximum Likelihood Estimator

One reason to turn to the maximum likelihood estimator (MLE) is that there exist cases where a Minimum Variance Unbiased Estimator either does not exist or cannot be found. In particular, this is true when the estimator does not satisfy Eq. 3.33. An example of such an estimator is given by Kay [6], where the signal model is

$$x_i = A + n_i, \tag{3.53}$$

where A is an unknown DC level and the likelihood function is given by

$$p(\mathbf{x}|A) = \frac{1}{(2\pi A)^{N/2}} e^{-\frac{1}{2A}\sum_{i=0}^{N-1}(x_i - A)^2}. \tag{3.54}$$

This is a somewhat curious example, since the unknown and the variance are the same, but it nevertheless serves as an example of an estimation problem for which an MVU cannot be found. That is, it cannot be put into the form of Eq. 3.33. Although an MLE can be found for this example, it is not efficient. This is true for many MLE estimators, i.e., that they are generally only asymptotically efficient and unbiased.

There is another reason we may need to turn to an MLE. It is that the MVU is based on a linear model, but most of the real-world problems that we will encounter here are nonlinear. However, the MLE has an important property expressed by the maximum likelihood invariance theorem [12]. As an example of this, suppose we seek the ML estimate of some parameter β. Further, suppose there exists another

parameter α, where there exists an (invertible) relationship between β and α, and the ML estimate of α is easy to determine. Since

$$\beta = f(\alpha). \tag{3.55}$$

and

$$\frac{\partial L}{\partial \beta} = \frac{\partial L}{\partial \alpha}\frac{\partial \alpha}{\partial \beta} = 0. \tag{3.56}$$

We see that

$$\frac{\partial L}{\partial \alpha} = 0, \tag{3.57}$$

satisfies the conditions for the ML estimate of both α and β. Thus, denoting $\hat{\beta}$ and $\hat{\alpha}$ as the ML estimates of β and α, respectively, it follows that

$$\hat{\beta} = f(\hat{\alpha}). \tag{3.58}$$

This property of the MLE is quite important for the problems encountered in this book. This is because as shown in [6], if an efficient estimator exists, it is given by the MLE. As an example of this, consider the general Gaussian case with a linear model. The likelihood function for this case is

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{(2\pi)^{N/2}|R|^{1/2}} e^{-\frac{1}{2}(\mathbf{y}-A\mathbf{x})^T R^{-1}(\mathbf{y}-A\mathbf{x})}. \tag{3.59}$$

Obviously, the maximum of $p(\mathbf{y}|\mathbf{x})$ obtains from the minimum of $(\mathbf{y}-A\mathbf{x})^T R^{-1}(\mathbf{y}-A\mathbf{x})$, which, as we have seen, delivers the MVU.

As will be seen, there are problems where a solution for parameters that are not of direct interest are more tractable, but these parameters are functions of the parameters of interest. Then the problem can be done in two steps: solve for those parameters not of direct interest followed by solving these estimates for the parameters of interest. It may turn out that a numerical solution will be necessary, but this is the price to pay for finding an ML solution, where a direct solution was simply not tractable.

When $p(y|x)$ is known, the ML type of estimator can be used. The advantage of this is that it is reasonably easy to find numerical solutions by searching for an extremum of the likelihood function, and in the case of a nonlinear model or a non-Gaussian pdf, this is a distinct advantage. For the case when no statistical information is available, the LS estimator is the only approach available, whether the model is linear or not. When the solution must be found numerically, a further complication here is that the likelihood surface may not be unimodal, i.e., there may be multiple extrema.

3.6 Bayesian Estimators

As mentioned in Chap. 1, Bayesian estimation [5] is the use of a priori information in an estimation. Formally, Bayes' rule states that

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}, \tag{3.60}$$

where $p(y|x)$ is the likelihood function as before, and $p(x|y)p(y) = p(x, y)$ is the joint probability. More will be said on this subject in the next chapter, but we wish to bring it up here in order to present a generalization of the CRLB that allows its use in the Bayesian problem. In Chap. 1 the maximum *a posteriori* (MAP) problem was introduced. Its variance was shown to be found from the addition of the Fisher information associated with the prior probability.[6] However, one may ask what the CRLB for the Bayesian estimation problem is in terms of the CRLB for the classical estimation problem. It is shown in [1] that it is found from the Fisher matrix resulting from the replacement of the likelihood function by the joint probability density function written as $p(y|x)p(x)$. The classical CRLB is based on the log likelihood and the Bayesian CRLB is based on the two terms, the log of the likelihood function and the log of the prior density function. This is seen as follows.

Consider the case of a single unknown parameter x. The log of the joint pdf is

$$ln(y, x) = lnp(y|x) + lnp(x). \tag{3.61}$$

The Fisher matrix then follows as

$$\mathbf{I} = -E\left\{\frac{\partial^2 lnp(y|x)}{\partial x^2}\right\} - E\left\{\frac{\partial^2 lnp(x)}{\partial x^2}\right\}. \tag{3.62}$$

So the problem separates neatly into the sum of two terms: the classical term and the prior term. As an example of an MAP problem, consider the following example.

We seek the estimate of the parameter θ, for the model

$$y_i = \theta + n_i, \quad i = 1, \quad 2, \quad \ldots, \quad N, \tag{3.63}$$

where n_i is Gaussian with covariance σ_{nn}^2.[7] Assuming θ to be a constant, we have

[6] The term Fisher Information is being used here in a loose manner. In the single parameter case, the reciprocal of the variance is referred to as the Fisher information whereas the inverse of the Fisher matrix, which obtains from the likelihood function, is the highest value attainable for the Fisher information.

[7] The double subscript is now used in anticipation of its conventional use in the Kalman filter literature.

$$ln\, p(\mathbf{y}|\theta) = -\frac{1}{2\sigma_{nn}^2}\sum_{i=1}^{N}(y_i - \theta)^2, \tag{3.64}$$

with $\mathbf{y} = [y_1,\ y_2,\ \ldots,\ y_N]$. This is an ML estimator and the solution follows from

$$\frac{\partial}{\partial\theta}\left\{\frac{1}{2\sigma_{nn}^2}\sum_{i=1}^{N}(y_i - \hat{\theta})^2\right\} = 0, \tag{3.65}$$

or

$$\hat{\theta}_{\text{ml}} = \frac{1}{N}\sum_{i=1}^{N} y_i. \tag{3.66}$$

That is, the estimate is the mean of the measurements. To find the variance on the estimate, the estimate is written as

$$\hat{\theta} = \frac{1}{N}\sum_{i=1}^{N} y_i = E\{\theta\} + \frac{1}{N}\sum_{i=1}^{N} n_i, \tag{3.67}$$

so that

$$\sigma_{\text{ml}}^2 = E[(\hat{\theta} - E\{\theta\})^2] = E\left\{\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} n_j n_i\right\} \tag{3.68}$$

since the noise is assumed to be uncorrelated, the double sum collapses to N times a single sum, so that the variance finally becomes

$$\sigma_{\text{ml}}^2 = \frac{1}{N}\sum_{i=1}^{N} E\{n_i^2\} = \frac{\sigma_{nn}^2}{N}. \tag{3.69}$$

Thus, the variance on the estimate of θ decreases as the number of data samples increases. This means that the estimator is consistent.[8]

Now suppose that there exists a priori information on θ that is Gaussian[9] with mean $\bar{\theta}$ and variance σ_{pp}^2. Then we seek the MAP solution

[8] A consistent estimator approaches the true value as the number of data samples increases.

[9] This is not a good example of a Bayesian estimator since, strictly speaking, the Bayesian prior information is considered to be stochastic information where here it is being used to perform a point estimation. Its use here is to present a simple means of illustrating the impact of prior information on an estimate.

$$\frac{\partial}{\partial\theta}\left\{\frac{\sum_{i=1}^{N}[y_i-\theta_{\text{map}}]^2}{2\sigma_{nn}^2}+\frac{[\theta_{\text{map}}-\bar{\theta}]^2}{2\sigma_{pp}^2}\right\}=0 \tag{3.70}$$

or

$$\frac{-\sum_{i=1}^{N}[y_i-\theta_{\text{map}}]}{\sigma_{nn}^2}+\frac{[\theta_{\text{map}}-\bar{\theta}]}{\sigma_{pp}^2}=0. \tag{3.71}$$

Solving for $\hat{\theta}_{\text{map}}$ results in

$$\hat{\theta}_{\text{map}}=\frac{\hat{\theta}_{\text{ml}}+\frac{\sigma_{nn}^2}{N\sigma_{pp}^2}\bar{\theta}}{1+\frac{\sigma_{nn}^2}{N\sigma_{pp}^2}}. \tag{3.72}$$

The variance on the MAP estimate is found as follows:

$$\sigma_{\text{map}}^2=E\{(\theta-\hat{\theta}_{\text{map}})^2\} \tag{3.73}$$

substituting for $\hat{\theta}_{\text{map}}$ from Eq. 3.72 and defining

$$R=\frac{\sigma_{nn}^2}{N\sigma_{pp}^2}. \tag{3.74}$$

Equation 3.73 becomes, after some manipulation,

$$(1+R)^2\sigma_{\text{map}}^2=E\{[(\theta-\bar{\theta})+(\theta-\hat{\theta}_{\text{ml}})]^2\}. \tag{3.75}$$

Using the fact that $E\{(\theta-\theta_{\text{ml}})\}=0$, Eq. 3.75 becomes

$$(1+R)^2\sigma_{\text{map}}^2=\sigma_{pp}^2R^2+\frac{\sigma_{nn}^2}{N}, \tag{3.76}$$

and from the definition of R, after some manipulation, we finally get

$$\sigma_{\text{map}}^2=\frac{\sigma_{pp}^2\frac{\sigma_{nn}^2}{N}}{\sigma_{pp}^2+\frac{\sigma_{nn}^2}{N}}. \tag{3.77}$$

But $\frac{\sigma_{nn}^2}{N}=\sigma_{\text{ml}}^2$, so that

$$\sigma_{\text{map}}^2=\frac{\sigma_{pp}^2\sigma_{\text{ml}}^2}{\sigma_{pp}^2+\sigma_{\text{ml}}^2}, \tag{3.78}$$

or

$$\frac{1}{\sigma^2_{\text{map}}} = \frac{1}{\sigma^2_{\text{ml}}} + \frac{1}{\sigma^2_{pp}}. \tag{3.79}$$

This is consistent with Eq. 3.62 which states that the Fisher information for the MAP estimator is the sum of the Fisher information for the ML estimator plus the Fisher information associated with the prior.

As mentioned in the last footnote, the MAP is not the best example of a Bayesian estimator. However, for the case of model-based processing, it is useful for pedagogical reasons. First, note that the contribution of the prior information is manifestly clear through the ratio R. When R is small compared with unity, the improvement is negligible. R can be small for two reasons: the variance on the prior is large, meaning it does not do a good job importing information on the estimate, or N is large, meaning that the number of measurements is so large that the ML estimate overwhelms the contribution by the prior. Another way to say this is that the Fisher information contributed by the prior is small compared to that provided by the ML estimator.

Another point that can be clarified by the MAP estimator is that it provides a simple example of the mismatch problem. This is the fact that bad prior information can actually degrade the estimate rather than improving it. This can be seen by inspection of Eq. 3.72. The MAP estimate is actually made up of a weighted sum of the ML estimate, $\hat{\theta}_{\text{ml}}$ and the mean of the prior, $\bar{\theta}$. This means that any error in $\bar{\theta}$ will introduce a bias to the MAP estimate. In fact, the prior will always introduce a correction to the ML estimate unless $\hat{\theta}_{\text{ml}} = \bar{\theta}$.

In the next chapter, when the Kalman-type recursive estimators are developed, the Bayesian approach will again be discussed in a somewhat different context.

3.7 Recursive Estimator Structures

The preceding section discussed that group of estimators sometimes referred to as *classical* estimators, since they deal with deterministic parameters. They also form the basis for batch type processes, since they are employed by first collecting all of the data, and then exercising the algorithm. However, there are certain classes of problems that are best dealt with by using recursive type processors. For example, if the statistics are not stationary, or the parameters of interest are changing in time, a batch process clearly cannot be optimal. As a means of redirecting attention from classical batch estimation to recursive estimation, two simple examples of recursive estimators are developed here and without proof, are put into the form of a Kalman filter.

3.7.1 Estimation of the Mean of a Growing Data Set

Consider the mean of N samples (measurements) $\{x_n\}$ of a fixed quantity x where the mean is to be taken as an estimation of the value of x. This is given as

$$\hat{x}_N = \frac{1}{N}\sum_{n=1}^{N} x_n, \tag{3.80}$$

where the hat indicates the estimate. We now ask whether adding a new measurement to the data set requires the whole data set to be averaged again, or if there is a simpler way to handle it. The answer is that there is a simpler way to handle it. This can be seen as follows. Equation 3.80 is rewritten as

$$\hat{x}_N = \frac{1}{N}x_N + \frac{1}{N}\sum_{n=1}^{N-1} x_n = \frac{1}{N}x_N + \frac{(N-1)}{N}\frac{1}{(N-1)}\sum_{n=1}^{N-1} x_n. \tag{3.81}$$

But this simply states that

$$\hat{x}_N = \frac{1}{N}x_N + \frac{(N-1)}{N}\hat{x}_{N-1}. \tag{3.82}$$

That is, the new average is equal to the old average times $(N-1)/N$ plus the new datum divided by the total number of data points. This can be rearranged to take the following form.

$$\hat{x}_N = \hat{x}_{N-1} + \frac{1}{N}(x_N - \hat{x}_{N-1}). \tag{3.83}$$

This is actually a highly simplified form of a Kalman filter. It is trivial for three reasons: there is no measurement noise included, x_N is found by its direct measurement, and the model is the proper one, meaning that the prescription (take the average) to achieve the estimate of x is correct.

To make it look more like a Kalman filter, a new notation is now introduced by rewriting Eq. 3.83 as

$$\hat{x}(t|t) = \hat{x}(t|t-1) + K(t)\epsilon(t). \tag{3.84}$$

Here $\epsilon(t) = [x_N - \hat{x}(t|t-1)]$ is called the *innovation* and $K(t)$ is called the Kalman gain, which in this simple example is a constant equal to $\frac{1}{N}$. x_N is the Nth measurement. The notation $x(t|t-1)$ is meant to indicate the estimate of x predicted at time t based on the data taken up to and including time $t-1$. This is called the predicted state in Kalman filter parlance and $x(t|t)$ is called the corrected state, where the correction term is seen to be the innovation times the Kalman gain.

Note that the correction includes the new measurement. Simply stated, the Kalman filter recursive estimator provides a new estimate as the sum of the previous estimate plus a correction term based on the new measurement.

Although not proved in this chapter, the Kalman gain derives from the fact that the Kalman filter is an MMSE.

3.7.2 Further Generalizations

Suppose that x, the parameter of interest, was not directly observable, but was related to the measurement in a linear way and was corrupted with additive noise. That is, suppose that the actual measurement is denoted by y and that

$$y = Cx + v, \tag{3.85}$$

where v is zero mean white and Gaussian. Then the innovation would take the form

$$\epsilon(t) = [y_N - C\hat{x}(t|t-1)]. \tag{3.86}$$

In the case of an exact model, the innovation sequence evolving from the recursion would then simply be $v(t)$,[10] the measurement noise. The importance of this is that in the case of incorrect or incomplete models, which we will discuss later, the innovation sequence provides a test of the goodness of the model based on the deviation of the innovations sequence from being zero mean and white. This is one of the powerful properties of the Kalman filter; its ability to constantly monitor the fidelity of the model.

It is now seen that the recursive estimator now must be specified by two equations, The first is Eq. 3.84 which we generalize here, without proof, to a multichannel form, without the correction term, but with an additive system noise term, W. In these equations, X is the M-dimensional state vector, H is the $M \times M$-dimensional state transition matrix, Y is the N-dimensional measurement vector, and C is the $N \times M$ measurement matrix. Thus, the state equation is

$$X(t|t) = HX(t|t-1) + W(t), \tag{3.87}$$

and the measurement equation, a generalized form of Eq. 3.85, is

$$Y(t) = CX(t) + V(t). \tag{3.88}$$

Here, $W = [w_1 \ \cdots \ w_M]^T$ and $V = [v_1 \ \cdots \ v_N]^T$. Equations 3.87 and 3.88 are considered to be in the Gauss–Markov form.

[10] In the following development, time dependence will not always be made explicit, in the interest of simplicity.

The term state equation is used since the original form of the filter was based on a state space-based formulation so as to accommodate differential equations. As an example, we consider a dynamic model based on a familiar dynamical system governed by the following second-order ordinary differential equation.

$$\frac{d^2x}{dt^2} + k^2x = 0. \tag{3.89}$$

Following the usual state space prescription, we define $\frac{dx}{dt} = \dot{x}$, and the two-dimensional state vector as $X = [x \quad \dot{x}]^T$. For this case, the Kalman equations, Eqs. 3.87 and 3.88 become, respectively,

$$\frac{d}{dt}\begin{bmatrix} x(t|t) \\ \dot{x}(t|t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k^2 & 0 \end{bmatrix}\begin{bmatrix} x(t|t-1) \\ \dot{x}(t|t-1) \end{bmatrix} + \begin{bmatrix} w_1(t-1) \\ w_2(t-1) \end{bmatrix}, \tag{3.90}$$

and

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = C\begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} v_1 \\ \vdots \\ v_N \end{bmatrix} \tag{3.91}$$

here C is the $N \times M$ measurement matrix. This is the linear Kalman filter and the equations are referred to as being in a first-order Gauss–Markov form.[11]

The solution to this Kalman filter estimator must be put into finite difference form before proceeding to a discrete recursive solution. Thus, we set

$$\frac{d}{dt}\begin{bmatrix} x(t|t) \\ \dot{x}(t|t) \end{bmatrix} \approx \frac{1}{\Delta t}\begin{bmatrix} x(t|t) - x(t|t-1) \\ \dot{x}(t|t) - \dot{x}(t|t-1) \end{bmatrix}. \tag{3.92}$$

After some manipulation, and assuming the approximation to be exact, the state equation becomes

$$\begin{bmatrix} x(t|t) \\ \dot{x}(t|t) \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ -\Delta t k^2 & 1 \end{bmatrix}\begin{bmatrix} x(t|t-1) \\ \dot{x}(t|t-1) \end{bmatrix}, \tag{3.93}$$

or

$$X(t|t) = [I + \Delta t H]X(t|t-1). \tag{3.94}$$

[11]The first-order Markov assumption states that in a discrete sequential system the present state depends only upon its previous value.

Defining $A = I + (\Delta t)H$, we have the resulting state equation

$$X(t|t) = AX(t|t-1), \tag{3.95}$$

and Eq. 5.17, which we repeat here, is the measurement equation, which completes the set of desired Kalman filter equations for this example.

$$Y(t) = CX(t). \tag{3.96}$$

This constitutes the complete Kalman filter equations for a discrete time solution for this second-order dynamical system. Given an a priori value for k in Eq. 3.89, this will deliver a sine wave solution. The accuracy of the solution will depend on the sample rate, $1/\Delta t$. The sufficiency of the size of Δt can be monitored by using the innovations sequence test for zero mean and whiteness. That is, Δt would be reduced until the innovations test is satisfactory.

This example is a demonstration of how the Kalman filter can accommodate physical models. The importance of this cannot be understated, since it demonstrates how it provides a self-consistent way to provide an enhancement to the performance of a processor by using physical models as a form of a priori information. For more discussion of this point see Candy [3]

3.8 The Linear Kalman Filter Algorithm

3.8.1 Preliminary Comments

There are three steps to the algorithm. The first is the prediction of the new state and its use to predict the new measurement, which is needed by the innovation. The state estimation error covariance $\tilde{P}$ is also predicted since it is needed in the second step, which is the computation of the correction terms. Also needed are an initial value for the measurement noise covariance matrix, R_{vv}, an initial value for the system noise covariance matrix, R_{ww}, and an initial guess for the state vector, $X(t-1|t-1)$. As mentioned before, a priori knowledge of the k term in Eq. 3.89 is needed. However, if it is poorly known, it can in fact be included as a third term in the state vector along with the necessary modification to the state transition matrix, and estimated along with the state itself. This is referred to as *augmentation*.[12] The third step is the update or correction step, which completes the iteration.

[12]This shows how the Kalman filter can self-consistently update the model along with the estimation of the quantities of interest.

3.8.2 The Algorithm

Prediction

$$\hat{X}(t|t-1) = A\hat{X}(t-1|t-1)$$

$$\tilde{P}(t|t-1) = A\tilde{P}(t-1|t-1)A'(t-1) + R_{ww}$$

$$\hat{Y}(t|t-1) = C\hat{X}(t|t-1)$$

Correction Terms

$$Y(t) = \text{New Measurement}$$

$$\epsilon(t) = Y(t) - C\hat{X}(t|t-1)$$

$$R_{\epsilon\epsilon}(t) = C(t)\tilde{P}(t|t-1)C'(t) + R_{vv}$$

$$K(t) = \tilde{P}(t|t-1)C'R_{\epsilon\epsilon}^{-1}$$

Update or correction

$$\hat{X}(t|t) = \hat{X}(t|t-1) + K(t)\epsilon(t)$$

$$\tilde{P}(t|t) = [I - K(t)C]\tilde{P}(t|t-1)$$

Here, $R_{\epsilon\epsilon}$ is the innovation covariance. It should be noted that the Kalman algorithm usually includes a source term in the first prediction equation that is not used here in the interest of simplicity, and will not be needed in the applications that we will be treating in Chap. 5.

3.8.3 Discussion

Observe that there are two terms that enter into the update equation. They are the Kalman gain and the innovation. It is worth taking a closer look at these terms. The Kalman gain is given by

$$K(t) = \tilde{P}(t|t-1)C(t)'R_{\epsilon\epsilon}^{-1}(t). \tag{3.97}$$

The $\tilde{P}(t|t-1)$ term is the predicted state error covariance. If it is large, it acts to increase the value of $K(t)$, thereby increasing the influence of the measurement on the correction term via the innovations. That is, because of the larger state estimation

error, the model loses some influence and the algorithm begins to depend more on the measurements. Alternatively, if the innovation covariance is large due to a large measurement noise, then the innovation covariance increases, thereby acting to reduce the impact of the measurement term on the update and putting more trust in the model. In this way, we see how the correction term encompasses both the impact of the fidelity of the model and the impact of measurement noise.

As previously mentioned, the system noise, $W(t)$, is not a noise in the same sense as the measurement noise. It is a means of allowing for incompleteness of the model. However, it is still a valid noise in the Gauss–Markov sense. It is of critical importance in the Kalman filter, without which the processor will not properly converge. Here the power of the Gauss–Markov model is evident, since it permits the Kalman filter to perform with models of varying fidelity, such that a degradation in the model fidelity, rather than preventing a solution, still allows a solution, but with the penalty of slower convergence and larger error covariance. Such latitude is not found in classical estimators.

Another important element of the Kalman filter is the innovation sequence, since it allows a continuous monitoring of the performance of the algorithm. If the model is doing a good job of extracting information from the data, then the innovation sequence will be zero-mean and white, since it will be predominantly made up of the measurement noise. Conversely, if the model deviates from a reasonably faithful representation of the physics, the innovations sequence will indicate this. Indeed, the innovation sequence can be easily configured into a sequential likelihood test and therefore can serve as a detector which can indicate the loss of an important element of the model. In other words, it can efficiently perform as a sensitive change detector. More will be said in this regard in the last chapter.

As a final point, although the system noise allows for incompleteness in the model, it should be pointed out that incompleteness is not the same as model errors as may be introduced, for example, by incorrect parameters. When model parameters are incorrect or greatly in error, system noise does not do a good job of compensating for this. This is why the ability of the Kalman filter to allow model parameters to be directly included in the estimation process, referred to earlier as *augmentation*, is so important. In other words, it solves the joint state/parameter estimation problem [4].

References

1. Bell, K.L., Van Trees, H.L.: Posterior Cramer-Rao bound for tracking target bearings. In: Proceedings of the Adaptive Sensor Array Processing Workshop, MIT Lincoln Labs, 7–8 June 2005
2. Burdic, W.S.: Underwater Acoustic System Analysis, p. 366, 2nd edn. Prentice-Hall, New Jersey (1991)
3. Candy, J.V.: Model-Based Signal Processing. Wiley, Hoboken (2006)
4. Candy J.V.: Bayesian Signal Processing: Classical, Modern, and Particle Filtering Methods. Wiley, New York (2009)

5. Frieden, B.R.: Physics From Fisher Information. Cambridge University Press, Cambridge (1998)
6. Kay, S.M.: Fundamentals of Statistical Signal Processing; Vol I, Estimation Theory. Prentice-Hall New Jersey (1993)
7. Kay, S.M.: Fundamentals of Statistical Signal Processing; Vol II, Detection Theory. Prentice-Hall, New Jersey (1998)
8. Lewis, T.O., Odell, P.L.: Estimation in Linear Models, pp. 52–55. Prentice-Hall, New Jersey (1971)
9. Moore, E.H.: On the reciprocal of the general algebraic matrix. Bull. Am. Math. Soc. **26**, 394–395 (1920)
10. Penrose, R.A.: Generalized inverse for matrices. Proc. Camb. Philos. Soc. **51**, 406–413 (1955)
11. Scharf, L.L.: Statistical Signal Processing; Detection, Estimation, and Time Series Analysis. Prentice-Hall, New Jersey (1993)
12. Zehna, P.W.: Invariance of maximum likelihood estimators. Ann. Math. Stat. **37**(3), i (1966)

Chapter 4
From Bayes to Kalman

4.1 Introduction

The Kalman filter [10] was introduced in the preceding chapter as a generalization of a simple recursive processor. This was a "bottom up" point of view and was presented without proof. In this chapter it will be shown that the Kalman filter is a special case of a more general processing structure called a Bayesian filter (BF). That is, it is presented as a "top down" description. None of the material in this book will deal in depth with the Bayesian filter, but it is introduced for completeness and also to demonstrate that the Kalman filter logically derives from it and is therefore a Bayesian processor. It then follows that for the case where the measurement model and the system model are both linear, and the measurement noise and system noise are both Gaussian, the Kalman filter is optimum and therefore an optimum realization of a Bayesian filter.

The Bayesian filter is based on the probability density functions (pdf) describing the situation, and is the general optimal filter. However, in general it cannot be used in this form since, unlike the optimal Kalman filter where the pdfs are Gaussian, the pdfs for the Bayesian filter cannot be represented in a closed form in the general case. The best one can do is to use a discrete representation of the pdfs, which leads to the so-called Particle Filter [3, 6]. In this hierarchy, the Bayesian filter comes first, followed by the Kalman filter. This point of view is actually the reverse of the actual historical development, since the Kalman filter came first [10] and was followed by the particle filter [3].

E.J. Sullivan, *Model-Based Processing for Underwater Acoustic Arrays*,
SpringerBriefs in Physics, DOI 10.1007/978-3-319-17557-7_4

4.2 The Bayesian Filter: Preliminaries

The Bayesian filter is based on a recursive form of Bayes' rule and uses the Chapman–Kolmogorov (C–K) equation to transition the state. Before discussing the details however, some discussion is in order.

Bayes' rule can be obtained by recognizing that the joint probability of two vector random variables X and Y can be written in two ways, i.e.,

$$P(X, Y) = P(X|Y)P(Y) = P(Y|X)P(X), \tag{4.1}$$

so that

$$P(X|Y) = P(Y|X)\frac{P(X)}{P(Y)}. \tag{4.2}$$

In words, this states that the *posterior* is equal to the *likelihood* times the *prior* divided by the *evidence* or normalizing factor. The normalizing factor is given by the marginal[1]

$$P(Y) = \int P(Y|X)P(X)dX, \tag{4.3}$$

so that we finally have Bayes' rule given as

$$P(X|Y) = \frac{P(Y|X)P(X)}{\int P(Y|X)P(X)dX}. \tag{4.4}$$

This forms the basis for the development of the chain rule of probability.

In order to facilitate the discussion of the chain rule, it is necessary to specialize the notation to a specific set of joint variables indexed by t. From Eq. 4.1, it follows that

$$P(y(t), y(t-1), y(t-2)) = P(y(t), y(t-1)|y(t-2)) \times P(y(t-2)). \tag{4.5}$$

In the same way, the RHS of this can be expanded as

$$\begin{aligned} &P(y(t), y(t-1)|y(t-2)) \times P(y(t-2)) \\ &\quad = P(y(t)|y(t-1), y(t-2)) \times P(y(t-1)|y(t-2)) \times P(y(t-2)). \end{aligned} \tag{4.6}$$

Combining these two, we have

$$\begin{aligned} &P(y(t), y(t-1), y(t-2)) \\ &\quad = P(y(t)|y(t-1), y(t-2)) \times P(y(t-1)|y(t-2)) \times P(y(t-2)). \end{aligned} \tag{4.7}$$

[1] A marginal distribution of a multivariate distribution is the result of removing one or more of the variates by summing or integration. Thus $p(x) = \int p(x, y)dy$ is a marginal distribution.

This is a general form of the chain rule, and it is easy to see that it can be extended to as many random variables as is necessary. For our purposes, we will assume that the process is first-order Markov. In words, the first-order Markov assumption states that in a discrete sequential system, the present state depends only upon the preceding state. What this means mathematically is that if we define

$$Y_t = \{y(t), y(t-2), y(t-3), \ldots\} \tag{4.8}$$

which is not a vector but simply denotes the set of measurements up to $y(t)$, then the expression

$$P(y(t), Y_{t-1}) = P(y(t)|Y_{t-1}) \times P(Y_{t-1}), \tag{4.9}$$

reduces to

$$P(y(t), Y_{t-1}) = P(y(t)|Y_{t-1}) \times P(y(t-1)), \tag{4.10}$$

so that

$$P(y(t)|Y_{t-1}) = P(y(t)|y(t-1)). \tag{4.11}$$

With this assumption Eq. 4.7 reduces to

$$\begin{aligned} &P(y(t), y(t-1), y(t-2)) \\ &\quad = P(y(t)|y(t-1)) \times P(y(t-1)|y(t-2)) \times P(y(t-2)). \end{aligned} \tag{4.12}$$

Thus, the Markov assumption leads to a simpler and very useful form of this form of the chain rule.

The Chapman–Kolmogorov equation now follows as a marginal pdf based on the chain rule. It is given by

$$P(x(t)|Y_{t-1}) = \int P(x(t)|x(t-1)) \times P(x(t-1)|Y_{t-1}) dx(t-1). \tag{4.13}$$

4.3 The Bayesian Filter

The Bayesian filter is based on the Chapman–Kolmogorov equation and the recursive Bayes' rule, where the former takes on the role of the predictor or state transition and the recursive Bayes' rule takes on the role of the measurement and update. Thus, the process model is embedded in the C–K equation, and the measurement in the form of the likelihood equation and the update are embedded in the recursive Bayes' rule. Returning for a moment to the Kalman filter, we see that the measurement and update are carried out in separate steps, where in the Bayesian filter, these steps are inherent in the update step of the recursive Bayes' rule.

Whereas the Kalman filter, which is Gaussian based, estimates the conditional mean and covariance of the state vector, the Bayesian filter prediction equation estimates the posterior density $P(x(t))|Y_{t-1})$, and the Bayesian update equation corrects this posterior based on the new measurement, thereby providing $P(x(t)|Y_t)$. These pdfs represent the complete solution. For example, there is no requirement that the posterior density be unimodal, or symmetric, and the process model, $P(x(t)|x(t-1))$, need not be linear.

Thus, the Bayesian filter is made up of the Chapman–Kolmogorov equation, which will henceforth be referred to as the *prediction* equation given by Eq. 4.13 and the update equation which is the recursive form of Bayes' rule [6] and is given here as

$$P(x(t)|Y_t) = \frac{P(y(t)|x(t))P(x(t)|Y_{t-1})}{P(y(t)|Y_{t-1})}. \tag{4.14}$$

This is derived in [6].

Here, the LHS is the updated posterior. On the RHS, we find the posterior based on the measurements up to time $t-1$ times $P(y(t)|x(t))$, which is the likelihood function, i.e., the pdf associated with the new measurement and will be designated as $C(y(t)|x(t))$ to single it out. This is the general update equation. The term $P(x(t)|x(t-1))$ in the update equation is now replaced by $A(x(t)|x(t-1))$ in order to identify it as the process model. In summary then, the Bayesian filter equations, for our purpose here, are given in the following form: the prediction equation, which predicts the value of the posterior pdf at time t, based on the measurements up to and including time $t-1$, *viz.*,

$$P(x(t)|Y_{t-1}) = \int A(x(t)|x(t-1)) \times P(x(t-1)|Y_{t-1})dx(t-1), \tag{4.15}$$

and the update equation which updates the predicted posterior to its corrected form, based on the new measurements[2] up to and including time t. Thus

$$P(x(t)|Y_t) = \frac{C(y(t)|x(t))P(x(t)|Y_{t-1})}{P(y(t)|Y_{t-1})}. \tag{4.16}$$

4.3.1 The Particle Filter

As mentioned in the previous section the Bayesian filter is the complete solution for the posterior pdf representing the estimation problem. However, as given in the two equations above, it is not of much use, since we generally do not know the pdfs in any useful closed form. In order to reduce it to an operational form, the particle filter

[2]The new measurement is entered directly into $C(y(t)|x(t))$.

(PF) has been developed. It is a direct numerical formulation of the Bayesian filter and is based on a discretization of the problem using Monte Carlo [7] sampling. The particles are the discrete samples of the pdf. That is, for the probability density $p(x)$, N discrete samples of x, say $\{x^i\}\ i = 1 \cdots N$, and the associated weights, $p(x^i)\ i = 1 \cdots N$, constitute a representation of the value of $p(x_k)$. This leads to a discrete representation of the prior $P(x(t)|y(t-1))$ given by

$$P(x_{1:k}|y_{1:k-1}) \approx \sum_{i=1}^{N} w_{k-1}^i \delta(x_k - x_k^i), \tag{4.17}$$

where k is the time index.

The pdf will have regions that are of varying importance (weight), resulting in the necessity for a nonuniform distribution of samples. Thus, given an initial set of these particles for the value of the posterior density $P(x(t-1)|Y_{t-1})$, this set is propagated through the prediction equation, resulting in a new set of particles representing $P(x(t)|Y_{t-1})$. The next step is the update, using Eq. 4.16. Although this may seem to be straightforward, it is not. To blindly carry out a recursive process will lead to a phenomenon called degeneracy, where only a few particles will have any significance and the others will be weighted with insignificantly small weights. This is dealt with by resampling, using a technique called Importance Sampling [6] and the procedure is referred to as sequential importance sampling[3] or SIS. Importance sampling is a technique that relies on selecting samples that cluster in the important region of the pdf, thus ignoring insignificant particles. This leads to the second problem where only a few samples survive and therefore cannot provide a significant representation of the distribution. This is called *impoverishment*. This approach is referred to as sampling importance resampling (SIR) and is sometimes called the bootstrap filter [5]. It is the workhorse of present day particle filtering.

Problems remain however. In particular, even when SIR is used, the issue of *impoverishment* can arise since the SIR approach converts a few particles into many, thus reducing the diversity of the particles. This can be dealt with in many cases by using a larger value of process noise.

Nevertheless, the particle filter is a powerful technique, and in those cases that deal with multi-modal pdfs, whether nonlinear or not, it is the only game in town. Examples of particle filter applications to ocean acoustic problems can be found in Michelopoulou et al. [11, 12] and references therein.

[3]The idea of importance sampling goes back to the Manhattan project and was originated by Von Neumann and Hastings in order to apply the so-called Monte Carlo methods to high dimensional integrals.

4.3.2 Comments

As can be inferred from the previous discussion, the particle filter is an extremely powerful technique for dealing with the nonlinear and non-Gaussian cases. However, it is not easy to use for the novice. As a result it is wise to avoid its use when possible, since the Kalman filter is still a valid approach to problems where the non-Gaussianity and the nonlinearities are not severe and a direct estimate of the full posterior probability density is not sought. Since this book deals with parameter estimation problems and not the direct estimate of the posterior density function of the system, the particle filter will not be of further interest to us in this book.

4.4 The Kalman Filter

Here, we will develop the Kalman filter on a more rigorous basis, by showing that it follows as a *maximum a posteriori* (MAP) estimate based on the posterior pdf of a Bayesian filter.[4] It is important to recall at this time that the Bayesian and particle filters provide a recursive estimate of the posterior pdf. However, the Kalman filter is most often used as a parameter estimator. For our purposes here then, the simplest way to connect the two are to consider the Kalman filter as a MAP estimator of x where x is considered as a deterministic state vector whose statistics are described by the posterior pdf from a Bayesian filter. In this approach, the update equation of the Bayesian filter is viewed as a MAP estimator with the prior pdf being the previous value of the posterior. That is, $P(x(t)|Y_t)$ is considered the output of an MAP estimator with $C(y(t)|x(t))$ being the likelihood function and $P(x(t)|Y_{t-1})$ being the prior. The update equation is

$$P(x(t)|Y_t) = \frac{C(y(t)|x(t))P(x(t)|Y_{t-1})}{P(y(t)|Y_{t-1})}. \tag{4.18}$$

The LHS is the posterior. On the RHS is the previous value of the posterior, i.e., the posterior based on the measurements up to time $t-1$, designated Y_{t-1}, times $C(y(t)|x(t))$, which is the pdf associated with the new measurement. Here, $C(y(t)|x(t))$ corrects or updates the posterior using both the new measurement and the measurement prediction update in the denominator. This term in the denominator is called the *evidence* in Bayesian parlance.

It is now assumed that the pdfs are Gaussian. Using the conventional notation for the Gaussian, i.e.,

$$N(x, m, R) = ae^{(x-m)'R^{-1}(x-m)}, \tag{4.19}$$

[4]This section is somewhat tedious to follow and can be skipped on a first reading without affecting the understanding of the rest of the book.

with x being a random vector, m its mean, a the normalization, and R the associated covariance matrix. For the Gaussian case then, the evidence from the denominator of the Bayesian filter takes the form of the following Gaussian density function:

$$P(y(t)|Y_{t-1}) = N(y(t), y(t|t-1), R_{\epsilon\epsilon}), \tag{4.20}$$

where R_{ee} is the innovations[5] covariance. The density associated with the measurement is

$$C(y(t)|x(t)) = N(y(t), Cx(t), R_{vv}) \tag{4.21}$$

with R_{vv} and C being the measurement noise covariance and measurement matrix, respectively.

$$P(x(t)|Y_{t-1}) = N(x(t), x(t|t-1), \tilde{P}(t|t-1)) \tag{4.22}$$

is the predicted posterior density, which is the pdf describing the statistics of the state vector estimate and $\tilde{P}(t|t-1)$ is the state error covariance matrix.

Some comments are due here. Note that, as was done in Chap. 3, the notation has been generalized somewhat by setting the time argument $(t-1)$ to $(t|t-1)$ where appropriate, to signify that at the time t, the state variable or measurement is based only on data up to $t-1$. Also note that Eq. 4.21 implies that the measurement equation is given by the Gauss–Markov form, i.e.,

$$y(t) = Cx(t) + v(t), \tag{4.23}$$

in keeping with our assumption of linearity. Thus, the innovation can be written as

$$\epsilon(t) = y(t) - C\hat{x}(t|t-1) = y(t) - \hat{y}(t|t-1), \tag{4.24}$$

where $y(t)$ and $\hat{y}(t|t-1)$ are the new measurement and the predicted measurement, respectively. The state error covariance is given by[6]

$$\tilde{P}(t|t-1) = E\{[x(t) - \hat{x}(t|t-1)][x(t) - \hat{x}(t|t-1)]'\}. \tag{4.25}$$

Substituting Eqs. 4.20–4.22 into the Bayesian processor update equation, Eq. 4.16, results in[7]

$$\begin{aligned} p(x(t)|Y_t) &= B \times \exp\big[-(1/2)v'(t)R_{vv}^{-1}v(t)\big] \\ &\quad \times \exp\big[-(1/2)\tilde{x}'(t|t-1)\tilde{P}^{-1}\tilde{x}(t|t-1)\big] \\ &\quad \times \exp\big[+(1/2)\epsilon'(t)R_{\epsilon\epsilon}^{-1}\epsilon(t)\big]. \end{aligned} \tag{4.26}$$

[5] Recall that the innovation is defined in Eq. 3.84.

[6] This is not to be confused with the posterior, which is the statistical description of the state vector x, where $\tilde{P}$ is the covariance associated with the state *error*.

[7] Here, we have changed the notation by using the prime instead of T to indicate the transverse operation in an attempt to render the notation somewhat cleaner.

Here $\tilde{x}(t|t-1) = x(t) - \hat{x}(t|t-1)$ and B is the normalization. Following the usual procedure, we consider the maximum of the log of $P(x(t)|Y_t)$. Note that the quadratic form associated with the innovation covariance $R_{\epsilon\epsilon}$ does not depend upon x, so only the quadratic forms associated with $\tilde{P}$ and R_{vv} need to be considered. Then the MAP estimate of x is found by finding the extremum of L where

$$\begin{aligned} L = &(y(t) - Cx(t))^{'} R_{vv}^{-1}(t)(y(t) - Cx(t)) \\ &+(x(t) - \hat{x}(t|t-1))'\tilde{P}^{-1}(t|t-1)(x(t) - \hat{x}(t||t-1)). \end{aligned} \tag{4.27}$$

Observe that Eq. 4.23 has been used.

In order to take the derivative, the vector gradient chain rule is used. It is given by

$$\nabla_x(a'b) = \nabla_x(a')b + \nabla_x(b')a. \tag{4.28}$$

Taking the gradient of Eq. 4.27 with respect to $x(t)$, setting it to zero and solving for X_{map} results in[8]

$$\hat{X}_{\text{map}} = [C'R_{vv}^{-1}(t)C + \tilde{P}^{-1}(t|t-1)]^{-1} \times [\tilde{P}^{-1}(t|t-1)\hat{x}(t|t-1) + C'R_{vv}^{-1}y(t)]. \tag{4.29}$$

The use of the matrix inversion lemma, given by

$$(A + BD')^{-1} = A^{-1} - A^{-1}B(I + D'A^{-1}B)^{-1}D'A^{-1}, \tag{4.30}$$

on the square bracket on the LHS of Eq. 4.29 with

$$A = \tilde{P}^{-1}(t|t-1), \quad B = C'R_{vv}^{-1}, \quad C = D', \tag{4.31}$$

results in

$$\begin{aligned} &[C^{'}R_{vv}^{-1}C + \tilde{P}^{-1}(t|t-1)]^{-1} \\ &= \tilde{P}(t|t-1) - \tilde{P}(t|t-1)C^{'}R_{vv}^{-1}[I + C\tilde{P}(t|t-1)C^{'}R_{vv}^{-1}]^{-1} \times C\tilde{P}(t|t-1). \end{aligned} \tag{4.32}$$

Before continuing, we need to have the innovations covariance in terms of the state estimation error covariance. The innovations covariance can be written as

$$R_{\epsilon\epsilon}(t) = E\{[y(t) - C\hat{x}(t|t-1)][y(t) - C\hat{x}(t|t-1)]'\}, \tag{4.33}$$

[8]This assumes that the maximum of the pdf occurs at X_{map}, which limits the validity of this approach to unimodal pdf's. Of course the Gaussian has this property.

and since $y(t) = Cx(t) + v(t)$ and $\tilde{x}(t|t-1) = x(t) - \hat{x}(t|t-1)$, this becomes

$$R_{\epsilon\epsilon}(t) = E\{[C\tilde{x}(t|t-1) + v(t)][C\tilde{x}(t|t-1) + v(t)]'\}. \tag{4.34}$$

Carrying out the expected value operation results in

$$R_{\epsilon\epsilon}(t) = C\tilde{P}(t|t-1)C' + R_{vv}(t). \tag{4.35}$$

Using Eq. 4.35 it is now possible to simplify the RHS of Eq. 4.32 by noting that the term in square brackets can be rewritten as follows:

$$[I + C\tilde{P}(t|t-1)C'R_{vv}^{-1}]^{-1} = [R_{vv}R_{vv}^{-1} + C\tilde{P}(t|t-1)C'R_{vv}^{-1}]^{-1}. \tag{4.36}$$

Now using Eq. 4.35, this reduces to $R_{vv}R_{\epsilon\epsilon}^{-1}$ so that the RHS of Eq. 4.32 reduces to

$$\tilde{P}(t|t-1) - \tilde{P}(t|t-1)C'R_{\epsilon\epsilon}^{-1}C\tilde{P}(t|t-1). \tag{4.37}$$

Substituting this into Eq. 4.29 results in

$$\begin{aligned}\hat{X}_{\text{map}} = &[\tilde{P}(t|t-1) - \tilde{P}(t|t-1)C'R_{\epsilon\epsilon}^{-1}C\tilde{P}(t|t-1)] \\ &\times[\tilde{P}^{-1}(t|t-1)\hat{x}(t|t-1) + C'R_{vv}^{-1}y(t)].\end{aligned} \tag{4.38}$$

Collecting terms in $\hat{x}(t|t-1)$ on the RHS of Eq. 4.38 results in

$$\hat{x}(t|t-1) - \tilde{P}(t|t-1)C'R_{\epsilon\epsilon}^{-1}C\hat{x}(t|t-1), \tag{4.39}$$

and collecting terms in $y(t)$ on the RHS of Eq. 4.38 yields

$$\tilde{P}(t|t-1)C'[R_{vv}^{-1} - R_{\epsilon\epsilon}^{-1}C\tilde{P}(t|t-1)C'R_{vv}^{-1}]y(t). \tag{4.40}$$

Using $I = R_{\epsilon\epsilon}^{-1}R_{\epsilon\epsilon}$ this becomes

$$\tilde{P}(t|t-1)C'R_{\epsilon\epsilon}^{-1}[R_{\epsilon\epsilon} - C\tilde{P}(t|t-1)C']R_{vv}^{-1}y(t), \tag{4.41}$$

and using Eq. 4.35, this term reduces to

$$\tilde{P}(t|t-1)C'R_{\epsilon\epsilon}^{-1}y(t). \tag{4.42}$$

Combining terms in $\hat{x}(t|t-1)$ and $y(t)$, Eq. 4.38 becomes

$$\hat{X}_{\text{map}} = \hat{x}(t|t) = \hat{x}(t|t-1) + K(t)\epsilon(t), \tag{4.43}$$

which is the Kalman state update equation.

Where

$$K(t) = \tilde{P}(t|t-1)C'R_{\epsilon\epsilon}^{-1}, \tag{4.44}$$

is the Kalman gain and

$$\epsilon(t) = y(t) - C\hat{x}(t|t-1), \tag{4.45}$$

is the innovation.

We are now in a position to derive the state estimation error covariance update. $\tilde{P}(t|t)$ is defined as

$$\tilde{P}(t|t) = E\{[\tilde{x}(t|t)][\tilde{x}(t|t)]'\} = E\{[x(t) - \hat{x}(t|t)][x(t) - \hat{x}(t|t)]'\}. \tag{4.46}$$

Using Eq. 4.43, this can be written as

$$\tilde{P}(t|t) = E\{[\tilde{x}(t|t-1) - K(t)\epsilon(t)][\tilde{x}(t|t-1) - K(t)\epsilon(t)]'\}. \tag{4.47}$$

Expanding this results in

$$\begin{aligned}\tilde{P}(t|t) = E\{&\tilde{x}(t|t-1)\tilde{x}'(t|t-1) - \tilde{x}(t|t-1)(K(t)\epsilon(t))' \\ &-K(t)\epsilon(t)\tilde{x}'(t|t-1) + K(t)\epsilon(t)(K(t)\epsilon(t))'\}.\end{aligned} \tag{4.48}$$

The first term is simply $\tilde{P}(t|t-1)$ and the last term can easily be shown to be

$$E\{K(t)\epsilon(t)(K(t)\epsilon(t))'\} = \tilde{P}(t|t-1)C'K(t)'. \tag{4.49}$$

Using the fact that

$$C\tilde{x}(t|t-1)\} = \epsilon(t) - v(t), \tag{4.50}$$

and the fact that the expected value of $v(t)$ is zero, the second term can be shown to be the negative of the third term, so that Eq. 4.48 reduces to

$$\tilde{P}(t|t) = \tilde{P}(t|t-1) - E\{K(t)\epsilon(t)(\tilde{x}'(t|t-1)\} \tag{4.51}$$

and again using Eq. 4.50, we finally have

$$\tilde{P}(t|t) = [I - K(t)C(t)]\tilde{P}(t|t-1), \tag{4.52}$$

which is the desired state error covariance update equation.

4.4.1 The Kalman Algorithm

Before we can show the complete algorithm, we must first derive the state error covariance prediction. For the linear Gaussian case, the state transition equation is based on the Gauss–Markov model given by

$$\hat{x}(t|t-1) = A(t-1)\hat{x}(t-1|t-1) + w(t-1), \tag{4.53}$$

where w is referred to here as the plant or system noise. The state estimation error is now

$$\tilde{x}(t|t-1) = x(t) - A(t-1)\hat{x}(t-1|t-1) - w(t-1), \tag{4.54}$$

or

$$\tilde{x}(t|t-1) = A(t-1)x(t-1) - A(t-1)\hat{x}(t-1|t-1) - w(t-1), \tag{4.55}$$

which is the same as

$$\tilde{x}(t|t-1) = A(t-1)\tilde{x}(t-1|t-1) + w(t-1). \tag{4.56}$$

The state error covariance prediction now follows from

$$\tilde{P}(t|t-1) = E\{\tilde{x}(t|t-1)\tilde{x}'(t|t-1)\}. \tag{4.57}$$

Substitution of Eq. 4.56 and performing the expected value operation leads to

$$\tilde{P}(t|t-1) = A(t-1)\tilde{P}(t-1|t-1)A'(t-1) + R_{ww}(t-1), \tag{4.58}$$

with $R_{ww}(t-1)$ being the system noise covariance. We now have all we need to specify the algorithm.

Prediction

$$\hat{x}(t|t-1) = A(t-1)\hat{x}(t-1|t-1) + B(t-1)u(t-1)$$
$$\tilde{P}(t|t-1) = A(t-1)\tilde{P}(t-1|t-1)A'(t-1) + R_{ww}(t)$$

Correction Terms

$$\epsilon(t) = y(t) - C(t)\hat{x}(t|t-1)$$
$$R_{\epsilon\epsilon}(t) = C(t)\tilde{P}(t|t-1)C'(t) + R_{vv}(t)$$
$$K(t) = \tilde{P}(t|t-1)C'(t)R_{\epsilon\epsilon}^{-1}(t)$$

Update

$$\hat{x}(t|t) = \hat{x}(t|t-1) + K(t)\epsilon(t)$$
$$\tilde{P}(t|t) = [I - K(t)C(t)]\tilde{P}(t|t-1)$$

The term $B(t-1)u(t-1)$ in the first prediction equation is a source term. As previously mentioned, we will not have occasion to use it in this book, but it is included here for completeness.

4.4.2 Some Comments

There are two terms that enter into the update equation. Here, they take the form of the Kalman gain and the innovation. It is worth taking a closer look at these terms. The Kalman gain is given by

$$K(t) = \tilde{P}(t|t-1)C'(t)R_{\epsilon\epsilon}^{-1}(t). \tag{4.59}$$

The $\tilde{P}(t|t-1)$ term is the state error covariance. If it is large, it acts to increase the value of $K(t)$, thereby increasing the influence of the measurement on the correction term via the innovations. That is, the model loses some influence and the algorithm begins to depend more on the measurements. Alternatively, if the innovation covariance is large due to a large measurement noise, then the innovation covariance increases, thereby acting to reduce the impact of the correction term on the update. In this way, we see how the correction term encompasses both the impact of the fidelity of the model and the impact of measurement noise.

The system noise $w(t)$ is a means of allowing for errors in the model. It is of critical importance in the Kalman filter, without which it will not properly converge. Here the power of the Gauss–Markov model is evident, since it permits the Kalman filter to perform with models of varying fidelity, such that a degradation in the model accuracy, rather than preventing a solution, allows a solution, but with the penalty of slower convergence and larger error covariance. Such latitude is rarely found in classical estimators.

4.4.3 The Nonlinear Case

As has already been mentioned, the Kalman filter is optimal only for the linear Gaussian case. Since many, if not most, real-world problems do not satisfy these criteria, approximations must be made. For the case of non-Gaussianity, as long as the problem is linear and the statistics are governed by a unimodal pdf, a direct application of the Kalman filter usually suffices. However, for the nonlinear case the

linear Kalman filter is not applicable. The first approach to deal with this was the so-called extended Kalman filter or EKF [4]. The EKF deals with nonlinearities by linearizing them with a first-order Taylor series approximation. This can be sufficient for many cases but can be difficult to use since the derivatives of the nonlinear functions can often be quite complex. Also they must be updated with each step.

In practice the EKF algorithm is nearly identical to the linear case algorithm, except that matrices of derivatives, in the form of Jacobian matrices, must be included for each nonlinear function, i.e., the state transition function and the measurement function. It will not be derived here and the derivation can be found in [4]. The algorithm is given by the following.

Prediction

$$\hat{x}(t|t-1) = A[\hat{x}(t-1|t-1)] + B(t-1)u(t-1)$$
$$\tilde{P}(t|t-1) = \mathscr{A}[\hat{x}(t-1|t-1)]\tilde{P}(t-1|t-1)\mathscr{A}[\hat{x}'(t-1|t-1)] + R_{ww}(t)$$

Correction Terms

$$\epsilon(t) = y(t) - C[\hat{x}(t|t-1)]$$
$$R_{\epsilon\epsilon}(t) = \mathscr{C}[\hat{x}(t|t-1)]\tilde{P}(t|t-1)\mathscr{C}[\hat{x}'(t|t-1)] + R_{vv}(t)$$
$$K(t) = \tilde{P}(t|t-1)\mathscr{C}[\hat{x}'(t|t-1)]R_{ee}^{-1}(t)$$

Update

$$\hat{x}(t|t) = \hat{x}(t|t-1) + K(t)\epsilon(t)$$
$$\tilde{P}(t|t) = [I - K(t)\mathscr{C}[\hat{x}(t|t-1)]\tilde{P}(t|t-1)$$

Jacobians

$$\mathscr{A}(x) = \left.\frac{\partial A[x]}{\partial x}\right|_{\hat{x}(t-1|t-1),\ u(t-1)} \qquad \mathscr{C}(x) = \left.\frac{\partial C[x]}{\partial x}\right|_{\hat{x}(t|t-1)}$$

Here, $A[x]$ and $C[x]$ denote the nonlinear state transition and measurement functions, respectively.

The term Jacobian is sometimes used to denote the matrix itself and other times it is used to denote its determinant. Here of course, it is the matrix. Also, as used here, it is not usually square.

4.4.4 The Unscented Kalman Filter

The EKF achieves its goal by linearizing the nonlinearity by means of a first-order Taylor expansion about a single point, whereas the unscented Kalman filter (UKF) uses the actual nonlinearity [8, 9], but seeks only the first two moments of the pdf. As we will see, it can provide an *exact* representation of the first two moments after transformation, when given the first two moments of the input in terms of the so-called *sigma points*. Further, it avoids the need for computing the Jacobians and in general is correct to second order and in the case of a Gaussian input it is correct to third order.

As an example, consider the case of a two-dimensional state vector with Gaussian noise with mean and covariance μ_x and R_{xx}. Further, R_{xx} is factorable into its square roots[9] as $R_{xx} = S_x S_x'$. The UKF is based on a set of points called sigma points, which are selected so as to capture the mean and covariance. It requires $2N_x + 1$ sigma points, where N_x is the size of the state vector. Thus, for this example, five sigma points are required. They are given by [9]

$$\begin{aligned}
\mathscr{X}_0 &= \mu_x & W_0 &= \frac{\kappa}{N_x + \kappa} \\
\mathscr{X}_1 &= \mu_x + \sqrt{(N_x + \kappa)}(S_x)_1 & W_1 &= \frac{1}{2(N_x + \kappa)} \\
\mathscr{X}_2 &= \mu_x + \sqrt{(N_x + \kappa)}(S_x)_2 & W_2 &= \frac{1}{2(N_x + \kappa)} \\
\mathscr{X}_3 &= \mu_x - \sqrt{(N_x + \kappa)}(S_x)_1 & W_3 &= \frac{1}{2(N_x + \kappa)} \\
\mathscr{X}_4 &= \mu_x - \sqrt{(N_x + \kappa)}(S_x)_2 & W_4 &= \frac{1}{2(N_x + \kappa)}.
\end{aligned} \tag{4.60}$$

Here, $(S_x)_1$ and $(S_x)_2$ are the two columns constituting the matrix S_x. The parameter κ is a tuning parameter. A heuristic rule commonly used is to set $N_x + \kappa = 3$.[10] The covariance is now represented in terms of the columns of its square root. Observe that for this example, since x is a vector of length two, each of these sigma points is a vector of length two. It is now easy to verify

$$\sum_{i=0}^{4} W_i \mathscr{X}_i = \mu_x, \tag{4.61}$$

and

$$\sum_{i=1}^{4} W_i (\mathscr{X}_i - \mu_x)(\mathscr{X}_i - \mu_x)' = R_{xx}, \tag{4.62}$$

[9] This factorization is usually done with a Cholesky decomposition.

[10] This parameter is discussed by Candy [6].

where we have used the fact that since $R_{xx} = S_x S_x'$, and $S_x = [(S_x)_1 \;\; (S_x)_2]$, then $[(S_x)_1 \;\; (S_x)_2][(S_x)_1 \;\; (S_x)_2]' = [(S_x)_1(S_x)_1' \;\; (S_x)_2(S_x)_2'] = R_{xx}$.

If these five points are now passed through a nonlinear state transition or measurement function $a[x]$, the result is the five new sigma points given by $\mathscr{Y}_k = a[\mathscr{X}_k]$. The new values of the mean and variance are now given by

$$\sum_{i=0}^{4} W_i \mathscr{Y}_i = \mu_y, \tag{4.63}$$

and

$$\sum_{i=1}^{4} W_i (\mathscr{Y}_i - \mu_y)(\mathscr{Y}_i - \mu_y)' = R_{yy}. \tag{4.64}$$

To gain some insight into this, it is illuminating to look at the scalar case. In this case, the initial state is specified by μ_x and σ_x^2.[11] The sigma points are given by

$$\begin{aligned}
\mathscr{X}_0 &= \mu_x \;\; = \;\; x(t-1|t-1) \\
\mathscr{X}_1 &= \mu_x + \sqrt{(N_x + \kappa)}\sigma_x \\
\mathscr{X}_2 &= \mu_x - \sqrt{(N_x + \kappa)}\sigma_x,
\end{aligned}$$

where here $N_x = 1$ and $\kappa = 2$ so that $N_x + \kappa = 3$. If the weights are defined as

$$\begin{aligned}
W_0 &= \frac{\kappa}{N_x + \kappa} \qquad = \frac{2}{3} \\
W_1 &= \frac{1}{2(N_x + \kappa)} \quad = \frac{1}{6} \\
W_2 &= \frac{1}{2(N_x + \kappa)} \quad = \frac{1}{6}
\end{aligned}$$

then

$$\mu_x = \sum_{i=0}^{2} W_i \mathscr{X}_i \;\text{ and }\; \sigma_x = \sum_{i=0}^{2} W_i (\mathscr{X}_i - \mu_x)^2. \tag{4.65}$$

These sigma points are now passed through the nonlinear transition $a[x]$ yielding

$$\begin{aligned}
\mathscr{Y}_0 &= a[\mathscr{X}_0] = a[\mu_x] \\
\mathscr{Y}_1 &= a[\mathscr{X}_1] = a[\mu_x + \sqrt{(N_x + \kappa)}\sigma_x] \\
\mathscr{Y}_2 &= a[\mathscr{X}_2] = a[\mu_x - \sqrt{(N_x + \kappa)}\sigma_x].
\end{aligned}$$

[11] For the scalar case we call R_{xx} σ_x^2 for reasons of convention.

It now follows that

$$\mu_y = \sum_{i=0}^{2} W_i \mathscr{Y}_i \quad \text{and} \quad \sigma_y = \sum_{i=0}^{2} W_i (\mathscr{Y}_i - \mu_y)^2, \tag{4.66}$$

where these are the exact moments and the weights are the same as before. To see why this works, we compute the moments μ_y and σ_y directly by using a three-point Gaussian–Hermite (G–H) [1] quadrature integration. The three-point G–H integration of a function, say $f(x)$, is given by[12]

$$\int_{-\infty}^{\infty} e^{-x^2} f(x) dx = \sum_{i=1}^{3} w_i f(\xi_i) \tag{4.67}$$

with

$$w_1 = \frac{2}{3}\sqrt{\pi}, \quad w_2 = \frac{1}{6}\sqrt{\pi}, \quad w_3 = \frac{1}{6}\sqrt{\pi} \tag{4.68}$$

and

$$\xi_1 = 0, \qquad \xi_2 = \frac{1}{2}\sqrt{6}, \qquad \xi_3 = -\frac{1}{2}\sqrt{6}. \tag{4.69}$$

When $f(x)$ is expressible as a polynomial of order $2n-1$, the quadrature is exact. For this case this means that if $f(x)$ is expressible as a polynomial of order 5, it is exact, since for our three-point quadrature $n = 3$. Thus, we now seek the expectation of $y = a[x]$ where x is distributed as $N(\mu_x, \sigma_x^2)$. Then

$$E\{y\} = \frac{1}{\sqrt{2\pi}\sigma_x} \int_{-\infty}^{+\infty} a[x] e^{-\frac{(x-\mu_x)^2}{2\sigma_x^2}} dx. \tag{4.70}$$

Setting $z = \frac{x-\mu_x}{\sqrt{2}\sigma_x}$ so that $x = \sqrt{2}\sigma_x z + \mu_x$ and $dx = \sqrt{2}\sigma_x dz$, we find that

$$E\{y\} = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} e^{-z^2} a[\sqrt{2}\sigma_x z + \mu_x] dz, \tag{4.71}$$

which is given by G–H quadrature as

$$E\{y\} = \frac{1}{\sqrt{\pi}} \sum_{i=1}^{3} w_i a[\sqrt{2}\sigma_x \xi_i + \mu_x]. \tag{4.72}$$

[12] W_i is used for the UKF weight and w_i is used for the Gauss–Hermite weight.

Inserting the G–H parameters results in

$$E\{y\} = \frac{1}{\sqrt{\pi}}\left[\frac{2}{3}\sqrt{\pi}a[\mu_x] + \frac{1}{6}\sqrt{\pi}a[\mu_x + \sqrt{3}\sigma_x] + \frac{1}{6}\sqrt{\pi}a[\mu_x - \sqrt{3}\sigma_x]\right]. \quad (4.73)$$

This can be written as

$$E\{y\} = [W_1\mathscr{Y}_1 + W_2\mathscr{Y}_2 + W_3\mathscr{Y}_3] = \mu_y \quad (4.74)$$

which is the exact mean based on the transformed sigma points. The transformed variance now follows immediately as

$$E\{(y-\mu_y)^2\} = \frac{1}{\sqrt{2\pi}\sigma_x}\int_{-\infty}^{+\infty}[a[x]-\mu_y]^2 e^{-\frac{(x-\mu_x)^2}{2\sigma_x^2}}dx. \quad (4.75)$$

Using the same transformation of variables as before and evaluating the integral vis G–H quadrature, we find

$$E\{(y-\mu_y)^2\} = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{+\infty} e^{-z^2}[a[\sqrt{2}\sigma_x z + \mu_x] - \mu_y]^2 dz. \quad (4.76)$$

Which, again using G–H quadrature, becomes

$$\sigma_y^2 = \sum_{i=1}^{3} W_i(\mathscr{Y}_i - \mu_y)^2. \quad (4.77)$$

The importance of this development is that even if the sigma points are passed through a nonlinearity, the correct first and second moments will obtain from the same weights and transformed sigma points, since these expected values are computed with the same Gauss–Hermite parameters. This means that given some nonlinear transfer function, say $a[x]$, the first and second moments of the transformed points follow directly from replacing $\mathscr{X}_0$, $\mathscr{X}_1$ and $\mathscr{X}_2$ with $\mathscr{Y}_0 = a[\mathscr{X}_0]$, $\mathscr{Y}_1 = a[\mathscr{X}_1]$ and $\mathscr{Y}_2 = a[\mathscr{X}_2]$.

The is a powerful result since an essentially exact representation of the first two moments is obtained, in spite of the existence of the nonlinearity. Further, this approach is not limited to the first two moments, as can be seen from Eq. 4.70, since the form of the function $f(x)$ is essentially arbitrary.

Summarizing, given the first and second moments of the input to a nonlinear transfer function μ_x and σ_x^2, the sigma points and weights are given by Eq. 4.60. Passing these sigma points through the nonlinearity yields a new set of sigma points $\mathscr{Y}_0$, $\mathscr{Y}_1$ and $\mathscr{Y}_2$ such that the output statistics follow from

$$\mu_y = \sum_{i=0}^{2} w_i\mathscr{Y}_i \qquad \sigma_y^2 = \sum_{i=0}^{2} w_i(\mathscr{Y}_i - \mu_y)^2. \quad (4.78)$$

This is not a general proof since the Gauss–Hermite (G–H) approach outlined above is only equivalent to the UKF approach for the scalar case. The UKF requires $2N_x + 1$ points whereas the three-point G–H approach requires 3^{N_x} points. This means that as the dimension of the state vector increases, the number of points required by the G–H approach increases faster that for the UKF approach. This is demonstrated in the following, where the two-dimensional G–H case is developed.

The generalization of Eq. 4.70 for the two-dimensional case begins by seeking the expected value of the two-dimensional state vector $\mathbf{y}$.

$$E\{\mathbf{y}\} = \frac{1}{2\pi|S'|}\int_{-\infty}^{\infty} \mathbf{y} e^{-\frac{1}{2}(\mathbf{y}-\bar{\mathbf{y}})' R_{yy}^{-1}(\mathbf{y}-\bar{\mathbf{y}})} d\mathbf{y}. \tag{4.79}$$

Here, $R_{yy} = SS'$ is the covariance of $\mathbf{y} - \bar{\mathbf{y}}$ where $\mathbf{y} = [y_1\ y_2]'$ and S is the square root of R_{yy}. In order to continue, we need to put this into reduced form. Following Arasaratnam et al. [2] with some minor changes, the following transformation of variables is introduced.

$$\mathbf{y} = \sqrt{2}S\mathbf{x} + \bar{\mathbf{y}}. \tag{4.80}$$

Observing that $\nabla_{\mathbf{y}}\mathbf{x}$ is the Jacobian, the resulting integral for the expected value of $\mathbf{y}$ is

$$E\{\mathbf{y}\} = \frac{1}{\pi}\int_{-\infty}^{\infty} \mathbf{y} e^{-\mathbf{y}'\mathbf{y}} \left[\sqrt{2}S\mathbf{x} + \bar{\mathbf{y}}\right] d\mathbf{x}. \tag{4.81}$$

Using the same three-point Gauss quadrature, but in nested form, this becomes

$$E\{\mathbf{y}\} = \sum_{i=1}^{3} w_i \sum_{j=1}^{3} w_j \left(\sqrt{2}[S_1\xi_i \ \ S_2\xi_j] + \bar{\mathbf{y}}\right), \tag{4.82}$$

where S_1 and S_2 are the columns of S. Note that this is already a nine-point case, compared to five points for the UKF. Expanding Eq. 4.82 and using the fact that $\xi_1 = 0$ results in

$$\begin{aligned}
E\{\mathbf{y}\} = {} & w_1^2\bar{\mathbf{y}} \\
& + w_1 w_2(\sqrt{2}[0 \ \ S_2\xi_2] + \bar{\mathbf{y}}) \\
& + w_1 w_3(\sqrt{2}[0 \ \ S_2\xi_3] + \bar{\mathbf{y}}) \\
& + w_2 w_1(\sqrt{2}[S_1\xi_2 \ \ 0] + \bar{\mathbf{y}}) \\
& + w_2^2(\sqrt{2}[S_1\xi_2 \ \ S_2\xi_2] + \bar{\mathbf{y}}) \\
& + w_2 w_3(\sqrt{2}[S_1\xi_2 \ \ S_2\xi_3] + \bar{\mathbf{y}}) \\
& + w_3 w_1(\sqrt{2}[S_1\xi_3 \ \ 0] + \bar{\mathbf{y}}) \\
& + w_3 w_2(\sqrt{2}[S_1\xi_3 \ \ S_2 S_2\xi_2] + \bar{\mathbf{y}}) \\
& + w_3^2(\sqrt{2}[S_1\xi_3 \ \ S_2\xi_3] + \bar{\mathbf{y}}).
\end{aligned}$$

Using Eqs. 4.68 and 4.69 to evaluate the coefficients w_i and the Gauss points ξ_i, it can be verified that the above weighted sum of these nine terms indeed delivers $\bar{\mathbf{y}}$, and after subtracting $\bar{\mathbf{y}}$, multiplying by the transpose, and performing the weighted sum, the correct covariance obtains. Further, it can be verified that the coefficients are normalized. In spite of the increase in the number of terms, the Gauss–Hermite approach does have some advantages and has been treated in depth by Arasaratnam et al. [2]. At this point, we leave the G–H approach and will use only the UKF method, which we summarize here.

$$\begin{aligned} \mathscr{X}_0 &= \mu_x & W_0 &= \frac{\kappa}{N_x + \kappa} \\ \mathscr{X}_i &= \mu_x + \sqrt{(N_x + \kappa)} S_i & W_i &= \frac{1}{2(N_x + \kappa)} \\ \mathscr{X}_{i+N_x} &= \mu_x - \sqrt{(N_x + \kappa)} S_i & W_{i+N_x} &= \frac{1}{2(N_x + \kappa)}. \end{aligned} \tag{4.83}$$

As previously mentioned, S_i is a column of the covariance square root.

4.4.5 The UKF Algorithm

Before discussing the algorithm per se there are two issues that must be clarified. These are the form of the Kalman gain and the issue of augmentation. First, the Kalman gain. As we already know, the Kalman gain for the linear case is given by

$$K(t) = \tilde{P}(t|t-1)C' R_{\epsilon\epsilon}^{-1}, \tag{4.84}$$

where C is the measurement matrix. For the UKF however, a measurement matrix per se is not available since the measurement is not linear. This means that the Kalman gain must be replaced by a form consistent with the nonlinear form of the UKF. In the following, using an approach used by Simon [13], with some changes, the necessary form for K is found by deriving the Kalman filter in a purely statistical form. This approach is illuminating since it does not explicitly specify Gaussian statistics, and assumes that the algorithm obtains from an assumption that the optimal filter is embodied in a linear operator which turns out to be the Kalman gain itself. Thus, the updated state is assumed to follow from

$$x(t|t) = K(t)y(t) + b(t). \tag{4.85}$$

Here, $y(t)$ is the latest measurement and $b(t)$ is a term that guarantees that the estimate is unbiased. That is, it is a constraint that requires

$$\bar{x}(t|t) = \bar{x}(t) = K(t)\bar{y}(t) + b(t), \tag{4.86}$$

where the overbar indicates the mean. Thus the constraint is

$$b(t) = \bar{x}(t) - K\bar{y}(t). \tag{4.87}$$

The value of K follows from the minimization of the trace[13] of $P(t|t)$, where

$$P(t|t) = E\{[x(t) - x(t|t)][\cdots]'\}. \tag{4.88}$$

We begin by subtracting and adding back the mean of $x(t) - x(t|t)$, resulting in

$$\begin{aligned} P(t|t) &= E\{[x(t) - x(t|t) - E(x(t) - x(t|t))][\cdots]'\} \\ &\quad + E\{[x(t) - x(t|t)]\}E\{[\cdots]\}'\}. \end{aligned} \tag{4.89}$$

Using Eqs. 4.85 and 4.87, the first term on the RHS of Eq. 4.89 can be written as

$$E\{[(x(t) - \bar{x}(t)) - K(t)(y(t) - \bar{y}(t))][\cdots]'\}. \tag{4.90}$$

Multiplying out and taking the expected value yields

$$\begin{aligned} &E\{[(x(t) - \bar{x}(t)) - K(t)(y(t) - \bar{y}(t))][\cdots]'\} \\ &\quad = P(t|t-1) - K(t)R'_{x\epsilon} - R_{x\epsilon}K'(t) + K(t)R_{\epsilon\epsilon}K'(t), \end{aligned} \tag{4.91}$$

where we have used the fact that $\epsilon = y(t) - \bar{y}(t) = y(t) - \hat{y}$.

The second term on the RHS of Eq. 4.89 is zero by virtue of Eq. 4.86. Hence,

$$P(t|t) = P(t|t-1) - K(t)R'_{x\epsilon} - R_{x\epsilon}K'(t) + K(t)R_{\epsilon\epsilon}K'(t). \tag{4.92}$$

The remaining task is to solve the following for $K(t)$.

$$\frac{\partial}{\partial K(t)}\text{Tr}[P(t|t)] = \frac{\partial}{\partial K(t)}\text{Tr}[-K(t)R'_{x\epsilon} - R_{x\epsilon}K'(t) + K(t)R_{\epsilon\epsilon}K'(t)] = 0, \tag{4.93}$$

where Tr is the trace operation. Note that $P(t|t-1)$ has been removed since it does not depend upon $K(t)$. The derivative follows from the development on page 308 in [13] which tells us

$$\frac{\partial}{\partial A}\text{Tr}(ABA') = AB + AB'; \quad \frac{\partial}{\partial A}\text{Tr}(AB) = B'; \quad \frac{\partial}{\partial A}\text{Tr}(BA') = B. \tag{4.94}$$

It now follows that by solving Eq. 4.93 for $K(t)$ yields

$$K(t) = R_{x\epsilon}R_{\epsilon\epsilon}^{-1}, \tag{4.95}$$

which is the Kalman gain in terms of the relevant covariance matrices.

[13] The trace is the sum of the squared errors, so its minimum is the minimum mean squared error.

This is still not the proper form of the gain for the UKF. To see why, it is useful to reexamine Eq. 4.84 by rewriting it in the following way.

$$K(t) = E\{\tilde{x}(t|t-1)\tilde{x}'(t|t-1)C'\}R_{\epsilon\epsilon}^{-1} \tag{4.96}$$

using the fact that $C\tilde{x}(t|t-1) = y(t) - \hat{y}(t|t-1) = \epsilon(t)$, it is immediately clear that this relation does not hold for the case of a nonlinear measurement function, since $\epsilon(t)$ is no longer a meaningful representation of the predicted measurement error. To remedy this, ϵ in the covariance matrices is replaced by the *residual* ξ_i, where

$$\xi_i = \mathscr{Y}_i(t|t-1) - \hat{y}(t|t-1), \tag{4.97}$$

with

$$\hat{y}(t|t-1) = \sum_{i=0}^{2N_x} W_i \mathscr{Y}_i(t|t-1). \tag{4.98}$$

The proper covariances are now given by

$$R_{\xi\xi}(t|t-1) = \sum_{i=0}^{2N_x} W_i \xi(t|t-1)\xi'(t|t-1), \tag{4.99}$$

and

$$R_{x\xi} = \sum_{i=0}^{2N_x} W_i \mathscr{X}_i(t|t-1)\xi'(t|t-1). \tag{4.100}$$

The second issue, that of augmentation, arises from the fact that in the case of nonadditive noise, the system noise and measurement noise covariances can be augmented directly into the state vector, thereby treating them as unknowns in the full nonlinear algorithm. There is a great deal of literature on this subject, and some disagreement as to its advantages. Here we will not be using it since in our experience, for the type of problems we deal with, additive noise seems to be a realistic assumption and augmentation does not seem to offer an advantage of any significance. For those interested, the reader is directed to the Ph.D. thesis of van der Merwe [14] and references therein.[14]

[14]This is a rather large and complete document on the subject of sigma-Point Kalman filters in general and is well worth reading.

4.4.6 A Walk Through the UKF Algorithm

For the case where the state function and the measurement function are both nonlinear, the nonlinear state transition and measurement functions are designated $A[x]$ and $C[x]$, respectively. The algorithm then proceeds as given in Table 4.1. Let the respective mean and covariance of the initial state be $\hat{x}(t-1|t-1)$ and $\tilde{P}(t-1|t-1)$. The sigma points in the first $2N_x+1$ lines in this table are then found from Eq. 4.83 with μ_x replaced by $\hat{x}(t-1|t-1)$ and S_p determined by $\tilde{P}(t-1|t-1) = S_p S_p'$. The associated weights also follow from Eq. 4.83.

The main advantages of this remarkable processor are [6]:

1. The transformed statistics are precise up to the second order.
2. The choice of the matrix square root used has no impact on the success of the algorithm.
3. No Jacobian calculations are necessary.

In the following chapter, several applications will be presented.

Table 4.1 The UKF algorithm

Sigma points and weights	
$\mathscr{X}_0 = \hat{x}(t-1\|t-1)$	$W_0 = \frac{\kappa}{N_x+\kappa}$
$\mathscr{X}_i = \hat{x}(t-1\|t-1) + \sqrt{(N_x+\kappa)}(S_p)_i$	$W_i = \frac{1}{2(N_x+\kappa)}$
$\mathscr{X}_{i+N_x} = \hat{x}(t-1\|t-1) - \sqrt{(N_x+\kappa)}(S_p)_i$	$W_{i+N_x} = \frac{1}{2(N_x+\kappa)}$
State prediction	
$\mathscr{X}_i(t\|t-1) = \mathbf{A}[\mathscr{X}_i(t-1\|t-1)] + \mathbf{B}[u(t-1)]$	(*sigma point prediction*)
$\hat{x}(t\|t-1) = \sum_{i=0}^{2N_x} W_i \mathscr{X}_i(t\|t-1)$	(*State Prediction*)
State error prediction	
$\tilde{\mathscr{X}}_i(t\|t-1) = \mathscr{X}_i - \hat{x}(t\|t-1)$	(*State Error*)
$\tilde{P}(t\|t-1) = \sum_{i=0}^{2N_x} W_i \tilde{\mathscr{X}}_i(t\|t-1)\tilde{\mathscr{X}}_i'(t\|t-1) + R_{ww}(t-1)$	(*Error Covariance*)
This completes the prediction stage of the algorithm. For the update stage, we use the predicted sigma points from the prediction stage corrected by the initial state error covariance.	
Update sigma points	
$\hat{\mathscr{X}}_0(t\|t-1) = \hat{x}(t\|t-1)$	
$\hat{\mathscr{X}}_i(t\|t-1) = \hat{x}(t\|t-1) + \sqrt{(N_x+\kappa)}(S_p)_i(t\|t-1)$	
$\hat{\mathscr{X}}_{i+N_x}(t\|t-1) = \hat{x}(t\|t-1) - \sqrt{(N_x+\kappa)}(S_p)_i(t\|t-1)$	
Measurement prediction	
$\mathscr{Y}_i(t\|t-1) = C[\hat{\mathscr{X}}_i(t\|t-1)]$	(*Nonlinear measurement*)
$\hat{y}(t\|t-1) = \sum_{i=0}^{2N_x} W_i \mathscr{Y}(t\|t-1)_i$	(*Measurement prediction*)

(continued)

Table 4.1 (continued)

Residual and gain prediction	
$\xi_i(t\|t-1) = \mathscr{Y}_i(t\|t-1) - \hat{y}(t\|t-1)$	(*Predicted residual*)
$R_{\xi\xi}(t\|t-1) = \sum_{i=0}^{2N_x} W_i \xi_i(t\|t-1)' \xi_i(t\|t-1) + R_{vv}$	(*Residual covariance*)
$R_{x\xi}(t\|t-1) = \sum_{i=0}^{2N_x} W_i \tilde{\mathscr{X}}_i(t\|t-1)' \xi_i(t\|t-1) + R_{vv}$	(*Cross covariance*)
$K(t) = R_{x\xi}(t\|t-1) R_{\xi\xi}^{-1}(t\|t-1)$	(*Gain*)
State update	
$\epsilon(t) = y(t) - \hat{y}(t\|t-1)$	(*Innovation*)
$\hat{x}(t\|t) = \hat{x}(t\|t-1) + K(t)\epsilon(t)$	(*State update*)
$\tilde{P}(t\|t) = \tilde{P}(t\|t-1) - K(t) R_{\xi\xi}(t\|t-1) K'(t)$	(*Error covariance update*)
Initial inputs	
$\hat{x}(0\|0) \quad \tilde{P}(0\|0) \quad R_{ww} \quad R_{vv}$	

References

1. Abramowitz, M., Stegun, I.A. (eds.): Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, 9th printing, p. 890. Dover, New York (1972)
2. Arasaratnam, I., Haykin, S., Elliot, R.J.: Discrete-time nonlinear filtering algorithms using Gauss-Hermite Quadrature. Proc. IEEE **95**(5), 953–977 (2007)
3. Arulampulam, M., Maskell, S., Gordon, T., Clapp, T.: A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian Tracking. IEEE Trans. Signal Process. **50**(2), 174–188 (2002)
4. Candy, J.V.: Model-Based Signal Processing. Wiley, Hoboken (2006)
5. Candy, J.V.: Bootstrap particle filtering. IEEE Signal Process. Mag. **74** (2007)
6. Candy, J.V.: Bayesian Signal Processing: Classical, Modern, and Particle Filtering Methods. Wiley, New York (2009)
7. Doucet, A., Godsill, S., Andrieu, C.: On sequential Monte-Carlo sampling for Bayesian filtering. Stat. Comput. **10**, 197–208 (2000)
8. Julier, S., Uhlmann, J.: A new extension of the Kalman filter to nonlinear systems. In: Proceedings of SPIE Conference, Orlando, FL (1997)
9. Julier, S., Uhlmann, J.: Unscented Kalman filtering for nonlinear estimation. Proc. IEEE **92**(3), 4001–4004 (2004)
10. Kalman, R.E.: A new approach to linear filtering and prediction problems. J. Basic Eng. Trans. Ser. D **82**(1), 35–45 (1960)
11. Michalopoulou, Z.-H., Jain, R.: Particle filtering for arrival time tracking in space and source localization. J. Acoust. Soc. Am. **132**, 3041–3052 (2012)
12. Michalopoulou, Z.-H,. Yardim, C., Gerstoft, P.: Particle filtering for passive fathometer tracking. J. Acoust. Soc. Am. **131**, EL74-EL80 (2012)
13. Simon, D.: Optimal State Estimation, pp. 318–320. Wiley, New York (2006)
14. van der Merwe, R.: Sigma point Kalman filters for probabilistic inference in dynamic state-space models. http://www.cslu.ogi.edu/publications/ps/merwe04.pdf

Chapter 5
Applications

5.1 Introduction

This chapter begins with the application of model-based processing to the towed line array for the case of a narrowband signal. It is the simplest of all of the examples and also clearly manifests the impact of explicitly using the sinusoidal signal configuration and the motion of the array as parts of the relevant model. Some time will be spent on this development so as to familiarize the reader with the details of the development of the UKF solution to the types of problems in this chapter. It will develop the processor as a bearing estimator. The outline of the code for this case will be presented in pseudocode with sufficient detail to allow the reader to develop his or her own MATLAB code. All problems demonstrated in this chapter have been solved using the UKF as described in Chap. 4.

The emphasis in the following examples will be on the improvement in performance achieved using the model-based approach over the conventional approach to the same problem. In most cases, the Cramér–Rao lower bound on the variance of the estimation error, as described in Chap. 3, will be used as the measure of improvement.

5.2 The Narrowband Towed Line Array

The model for this case has two parts. First, the motion of the array is explicitly included. Second, the functional form of the sinusoidal signal is explicitly used. The conventional approach to narrowband bearing estimation using a towed array, as described in Chap. 2, is to introduce a phase shift to the output of each receiver element, where this phase shift is consistent with a particular look direction. This is depicted in Fig. 5.1, where the phase shift is based on the $d\sin\theta$ term. The procedure

E.J. Sullivan, *Model-Based Processing for Underwater Acoustic Arrays*,
SpringerBriefs in Physics, DOI 10.1007/978-3-319-17557-7_5

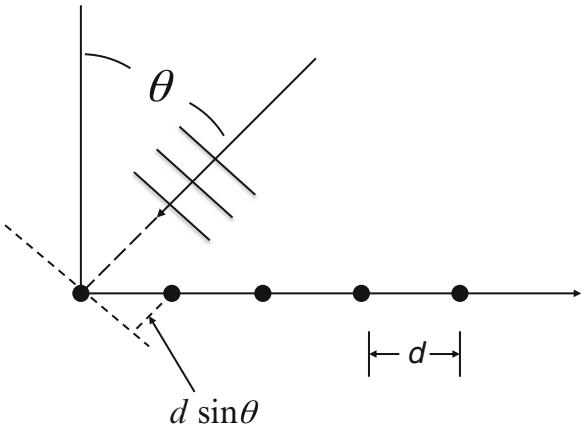

Fig. 5.1 Line array configuration

is then to sum these shifted receiver outputs producing the desired output of the beamformed array. In the model-based approach, rather than constructing such a beamformer, the problem is treated as a pure estimation problem.

Referring again to Fig. 5.1, note that the signal received at, say, the third receiver from the origin on the x-axis is located a distance of $x = 2d$ from the origin on the x-axis. The first element is labeled $n = 0$. This means that if we consider the signal at the receiver element at the origin as the reference signal, then the phase difference between the signal received at this element, and that received at element No. 2, is $2 \times 2\pi(d/\lambda) \times \sin\theta$, where θ is the angle of the incoming signal measured clockwise from the vertical (broadside). More generally, the relative phase of the signal at the nth element can be written as $nkd\sin\theta$ where we recall from Chap. 2 that $k = 2\pi/\lambda$ is called the wavenumber. Unlike the treatment in Chap. 2 however, where the temporal dependence was ignored and the problem was treated as a purely spatial problem, in the model-based method the time dependence cannot be ignored.

The approach is as follows. Referring to Fig. 5.2, the phase of the signal received at the hydrophone moving to the right with speed v is given by

$$\phi(t) = \omega_0[t + ([d + vt]/c)\sin\theta], \tag{5.1}$$

where ω_0 is the source frequency and the Doppler due to the motion is seen to be a function of the bearing angle θ. Generally speaking then, the phase at the nth element is

$$\phi_n(t) = \omega_0[t + ([nd + vt]/c)\sin\theta] = \omega_0[1 + (v/c)\sin\theta]t + nkd\sin\theta, \tag{5.2}$$

where the phase term now can be seen to include both the Doppler and the spatial phase term found in the usual nonmoving case. Although none of this is new, the fact remains that this Doppler term is still ignored in conventional bearing estimation schemes. Here we will show how the dependence of the Doppler on the bearing angle can be exploited to improve the bearing estimation performance.

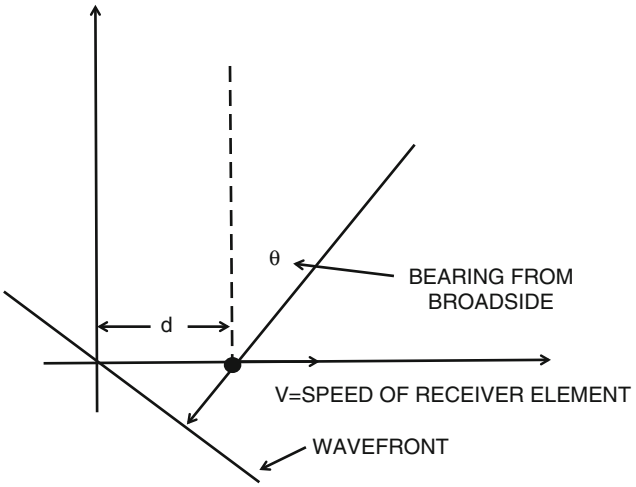

Fig. 5.2 The moving hydrophone. This depicts the second element, which has index $n = 1$, at $t = 0$

For the signal frequency of $\omega_0 = 2\pi f_o$, the phase of the signal for the stationary array at the nth element is $\omega_0 t + knd\sin\theta$. However, if the array is moving, say to the right at speed $+v$, then the correct expression for this phase term is given by Eq. 5.2. Here we see that although in the conventional approach, the observed (Doppler shifted) frequency is used when computing the spatial phase where, to be precise, the source frequency should be used. However, the conventional approach works well since the Doppler shift incurred by an array moving at a conventional tow speed is a small fraction of the source frequency. Nevertheless, since there is information about the bearing angle θ in the Doppler, it is worth asking if this information can be exploited in order to improve the bearing estimate.

In order to answer this question, consider a single element in Fig. 5.1 Since it is moving to the right with speed v, the observed radian frequency is given as before by

$$\omega = [1 + (v/c)\sin\theta]\omega_0. \tag{5.3}$$

Here, ω_0 is the source frequency.[1] Assuming v and c to be known, it can be seen that if the source frequency was known a priori, then the bearing angle θ could be found—even in the case of no physical aperture. That is, there is observable bearing information contained in the Doppler. Note that the array has nevertheless moved, tracing out a path $L = vT$, where T is the time assigned to the process. Thus, L can be considered to be a passive synthetic aperture, since the physical aperture is clearly zero.

[1] A more general form of Eq. 5.3 is $\omega = [1 \pm (v/c)\sin\theta]\omega_0$ where the minus sign allows for array motion in the $-x$ direction.

This argument suggests that if the bearing angle and source frequency were estimated jointly, then the quality of the bearing estimate would be improved. It was shown in 1997 by Sullivan and Candy [13] that this is indeed the case and that such an estimation can be carried out using a Kalman filter. One might ask if it is possible to perform a bearing estimation using a single hydrophone. The answer is no, and this point will be expanded upon later in this chapter.

5.2.1 Model-Based Bearing Estimation with a Towed Array

The measurement system for the bearing estimation problem is the set of outputs from the receiver elements of the array. Calling this measurement vector $Y = [y_1 \;\; y_2 \;\; \cdots \;\; y_N]'$, we have

$$\begin{aligned} y_1(t,0) &= a\cos[\omega_0 t + \beta(0,t)\sin\theta] \\ y_2(t,1) &= a\cos[\omega_0 t + \beta(1,t)\sin\theta] \\ &\vdots \\ y_M(t,N-1) &= a\cos[\omega_0 t + \beta(N-1,t)\sin\theta], \end{aligned} \tag{5.4}$$

where the argument of the cosine function is given by Eq. 5.2 along with the definition $\beta(n,t) = k(nd + vt)$, with n being the element index. This nonlinear measurement system can be expressed concisely as

$$Y = c[\theta, \omega_0, a]. \tag{5.5}$$

The state vector is $X = [\theta \;\; \omega_0 \;\; a]'$ and the state equation is

$$X(t|t-1) = \begin{bmatrix} \theta(t|t-1) \\ \omega_0(t|t-1) \\ a(t|t-1) \end{bmatrix} = AX(t-1|t-1) = \begin{bmatrix} 1\,0\,0 \\ 0\,1\,0 \\ 0\,0\,1 \end{bmatrix} \begin{bmatrix} \theta(t-1|t-1) \\ \omega_0(t-1|t-1) \\ a(t-1|t-1) \end{bmatrix}. \tag{5.6}$$

Some comments are in order here. Observe that the state equation contains three parameters instead of two. This is because the signal is expressed in terms of its amplitude and frequency and not its phase. By expressing the phase as a function of the two unknowns θ and ω_0, carrying the amplitude along as a nuisance parameter could be avoided. However, this necessitates employing a phase unwrapping step which can be problematical, especially when the SNR is low.

Note also that unlike most Kalman filter configurations, there is no structure to the state transition matrix A, i.e., it is simply the identity matrix. This is referred to as the random walk case and occurs here since the relevant model for this problem

is totally contained in the measurement system. However, the state transition matrix has been written explicitly here since later we will have occasion to include a bearing rate model which will necessarily provide structure to the A matrix. A bearing rate model will improve the bearing estimate for cases where the bearing rate is significant.

This problem, as formulated here, is the simplest example of the model-based bearing estimation problem. Indeed multiple bearings and frequencies can easily be introduced, but at the cost of a larger state vector. Also, the element spacings are not limited to be equal—any spacing scheme can be employed simply by replacing the nd term with the element coordinate x_n.

We are now in a position to configure the estimation algorithm, i.e., the construction of the unscented Kalman filter for this problem, since the state and measurement systems have been defined. This will be described in some detail for this problem so as to familiarize the reader with art of formulating model-based problem solutions using the UKF. The code will use an outer shell which requires an update function which in turn requires two functions for generating the sigma points and their weights.

The outer shell code, or driver program, will input the values of α, β, and κ,[2] the initial state vector $\hat{x}(t-1|t-1)$, N_x, and N_y, the respective sizes of the state and measurement vectors, and the initial values of the covariances $P0 = \tilde{P}(t-1|t-1)$, R_{ww} and R_{vv}, the initial state error covariance matrix, the state or system noise covariance matrix, and the measurement noise covariance matrix, respectively. These last three are referred to as the tuning[3] parameters, since they must be adjusted in order to optimize the convergence of the filter. The data file containing the input data time series from the receiver elements is also entered in this driver code. A **for** loop on the time is now put into the driver program which calls the update function. This update function will generate the sigma points as described under *sigma points and weights* in Table 4.1. The state prediction is then carried out as described under the second part of Table 4.1. Continuing with Table 4.1, the state error prediction is computed. Before performing the measurement prediction, the sigma points must be updated using the updated state mean and error covariance. The residual and the Kalman gain are then determined and finally the state is updated, completing the first iterative step. Also required to complete this step are two functions containing the state and measurement equations. This update function also contains the two functions tasked with computing the initial sigma points and

[2]Note that in Table 4.1, there is a term $N_x + \kappa$ that appears in several places. However, the user has a choice. The term κ can be replaced with $\lambda = \alpha^2(N_x + \kappa) - N_x$. The relative advantages or disadvantages of this are discussed by Candy [3] on pages 208–209, where the definitions of α and β can be found. The difference between the two manifests itself mainly in the higher order moments of the initial pdf of the problem. In Table 4.1 κ is used but in the code used here for this problem κ is replaced with λ. The reader is encouraged to try it both ways.

[3]Tuning is discussed in detail in Chap. 6.

their associated weights and the other to compute the updated sigma points for the measurement and update steps. Following the completion of this for loop, the driver program then must include the necessary plot routines.

This code structure is outlined by the following pseudocode[4]

Pseudocode

Input α, β, κ, R_{ww}, R_{vv}, $\tilde{P}(t-1|t-1)$, $P0$, Nf, $x(t-1|t-1)$, fs, vta

Load Data file This contains This contains

- Y(Nf,Ns)= Hydrophone input data time series—*Ns* is the number of time samples
- Nf=Number of hydrophones
- fs=Sample frequency
- $x(t-1|t-1)$=Initial state vector—this is 3×1 for our case, since $Nx = 3$
- $\tilde{P}(t-1|t-1) =$ Initial state error covariance
- vta=Towed array speed

For Loop

for (initial time) **to** (final time) **do**

call update function

This carries out calculations specified in Table 4.1, i.e.

- Compute sigma points and weights (done with function routines)
- State prediction
- State error prediction
- Update sigma points
- Measurement prediction
- Residual and gain prediction
- State update

end do

Output State vector estimates and other desired quantities

Plot Routines

5.2.1.1 Example

This example is that of a moving line array (or towed array) with five elements spaced at half-wavelength for the 121 Hz narrowband signal, arriving at the array at an angle of $+3°$ from broadside. The input data file contains the simulated data file for a 121 Hz signal of length 30 s generated as would be received by a moving array at a speed of $vta = 5\,\mathrm{m/s}$. That is, the motion is manifest in the data, since

[4]Pseudocode is an intermediate step between the common language and programming language, and is useful since it is logically structured, thus allowing the user to configure the actual code based on his or her personal experience while maintaining the proper form of the algorithm.

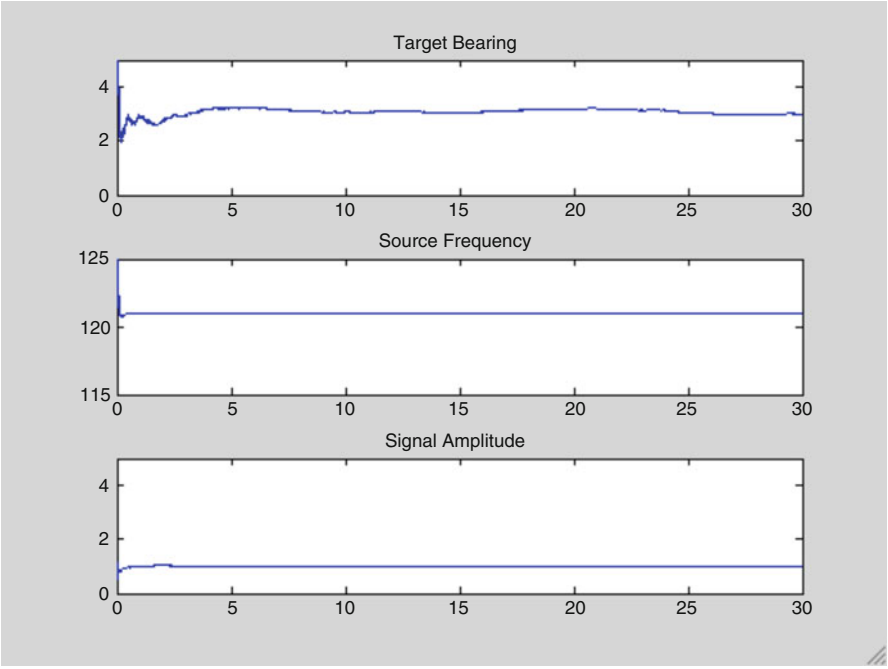

Fig. 5.3 Simulation results for a simulated signal arriving at a bearing of 3°. The simulated signal amplitude was set at unity and the source frequency is 121 Hz

the data file must reflect the fact that it is being received by a moving receiver. The SNR is −2 dB at the hydrophone level[5] and the noise was generated using the **randn** MATLAB function. Based on the previous discussion, the state vector is of length $Nx = 3$, The measurement vector is of length $N_y = 5$. The resulting tuning parameters are

$$P0 = \begin{bmatrix} 10\,0\,0 \\ 0\;1\;0 \\ 0\;0\,10 \end{bmatrix} \times 10^{-2}\ R_{ww} = \begin{bmatrix} 0.001 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 10 \end{bmatrix} \times 10^{-7}\ R_{vv} = \begin{bmatrix} 1\,0\,0 \\ 0\,1\,0 \\ 0\,0\,1 \end{bmatrix} \times 10^{-5}. \tag{5.7}$$

The initial state vector is $x0 = [30\ \ 125\ \ 1]' = [\theta\ \ \omega_0\ \ a]'$. The results are shown in Fig. 5.3.

Figure 5.4 shows the results of an experiment carried out in the Baltic sea [15] using the hydrophones on the forward end of a towed array. It was this scenario that was used as a basis for the above simulation. This figure shows the results of

[5]This SNR was picked so as to emulate as closely as possible the experiment carried out in the Baltic sea [15] described below.

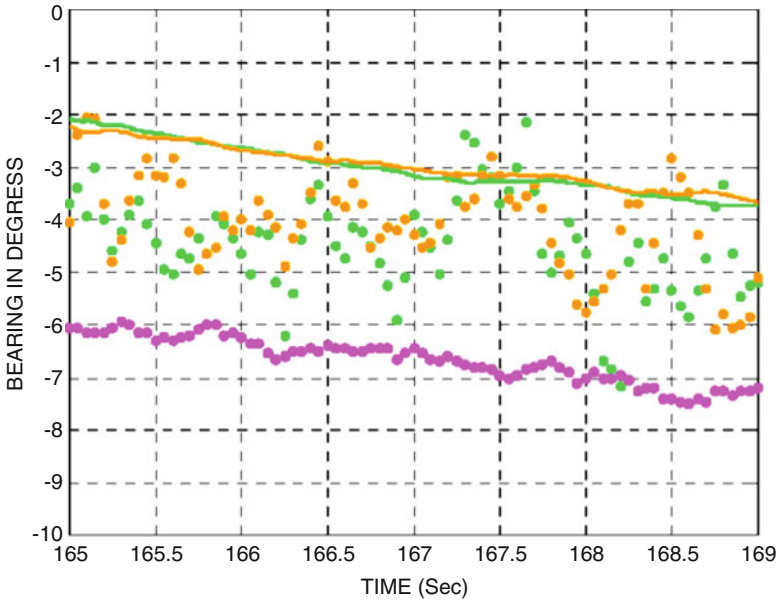

Fig. 5.4 Results of Baltic experiment

a 4 s record as the array passed near broadside to a pure tone source at a range of about 500 m. The full array was six wavelengths long at 121 Hz. The bottom curve is the result of using a conventional bearing estimator with the full aperture. The upper smooth curves are the result of using the forward section of the array both for the first four hydrophones (green) and the first five hydrophones (yellow). The simulation was for the five hydrophone case, which provides an aperture of about two wavelengths. The smooth curves are the results of using the model-based processor on the short forward sections and the widely scattered dots are the conventional bearing estimation results on the same forward sections. As can be seen, the model-based processor greatly outperforms the conventional processor. The offset between the upper curves and the lower curve is a consequence of the fact that the acoustic centers of the short forward sections and the full array are offset from each other, since the range to the source is only on the order of 500 m. The sample variance on the bearing estimate for the yellow dots, i.e., the two wavelength case, is $\sigma_B^2 = 0.234$. This is close to $\sigma_{fd}^2 = 0.196$ resulting from the use of a frequency domain beamformer (see the discussion of the $k - \omega$ beamformer in Chap. 3) on the simulated data shown in Fig. 5.5. The sample variance on the yellow curve in Fig. 5.4 is $\sigma_{\text{mb}}^2 = 0.0052$. Clearly, the impact of the model-based approach is enormous.

Finally, we show in Fig. 5.5 the result of using the frequency domain beamformer on the simulated data compared to the model-based result. As can be seen, these simulated results compare well to the experimental results. That is, the impact of the model-based approach is huge.

Fig. 5.5 Comparison of the bearing estimate for the model based processor (*top*) with that of the frequency domain beamformer (*bottom*)

5.2.2 The Single Hydrophone Case

As mentioned in Sect. 5.2 this algorithm cannot provide a bearing estimate while using a pressure measurement based on a single moving hydrophone. The question is not a trivial one, since the Kalman filter can provide estimates of state vectors where the size of the state vector is larger than the size of the measurement system. This follows since the number of measurements is determined not simply by the size of the measurement system, but by the fact that the processor is recursive, thereby providing a large number of measurements. This can be seen in the following development.

Consider the linear time-independent system given by

$$x(1) = Ax(0), \tag{5.8}$$

$$y(1) = Cx(1), \tag{5.9}$$

where x is 3×1 and y is 1×1.

By iterating the measurement system, the following structure follows.

$$y(0) = Cx(0)$$

$$y(1) = CAx(0)$$

$$y(2) = CA^2x(0)$$
$$\vdots$$
$$y(n-1) = CA^{n-1}x(0).$$

Or, since the state vector is 3×1, $n = 2$ and

$$\begin{bmatrix} C \\ CA \\ CA^2 \end{bmatrix} x(0) = M_o x(0) = \begin{bmatrix} y(0) \\ y(1) \\ y(2) \end{bmatrix}. \tag{5.10}$$

The matrix M_o is called the observability matrix and when it is full rank, the system is observable. This means that even though there are three unknowns, i.e., the three elements of the state vector, and only one measurement equation, that is, the measurement system is one-dimensional, the system is solvable. The analysis carried out above is based on a linear system. For the single-hydrophone case, since the measurement system is time-dependent and nonlinear, the above analysis does not apply. A proper analysis requires the use of the Gramian [3] and is a level of complexity beyond the scope of this book. However, when it is done, it concludes that the observability matrix is not full rank. For those interested in pursuing this further, [8] is a good start.

This concludes the discussion of the narrowband towed array bearing estimator. The next section generalizes the narrowband bearing estimation to the case of the joint estimation of bearing and range.

5.2.3 Joint Bearing and Range Estimation

Conventional tracking with a Kalman filter requires that a maneuver be made. This is done by making two bearing measurements, each at a sufficiently different bearing in order to insure that the measurements are independent. This is necessary since the conventional tracking algorithm uses separate bearing measurements that are made external to the Kalman filter, and then used as inputs, that is, it is a means of coherent triangulation and is used to provide an estimation of the position and velocity of the target. This is sometimes referred to as "bearings-only" tracking. For more on this see [1] and references therein.

This approach presents a problem if one desires a rapid estimation of the source coordinates, since to maneuver the township and then recover a straight line configuration of the towed array to obtain the second bearing estimate can require as long as 20 min. Here, we demonstrate that by using the model-based approach, where the motion of the array plays a direct role in the estimation process, the need for a maneuver can be eliminated. Here we will simply do a joint estimation where the range will be inferred by building the wavefront curvature into the measurement model.

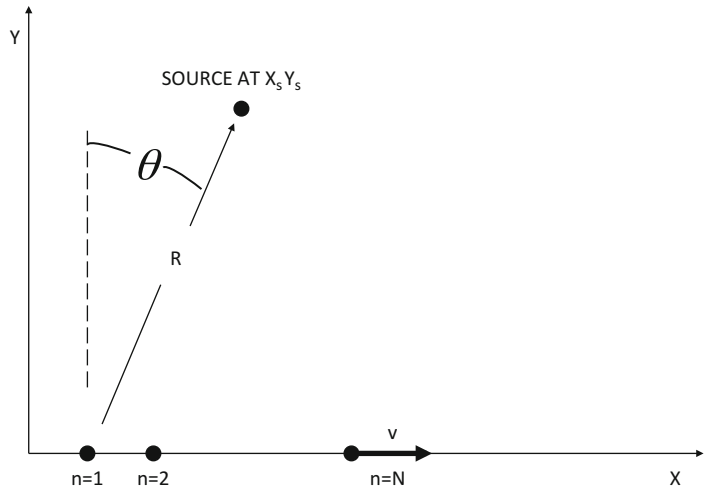

Fig. 5.6 Array and source configuration. R is the range to the first element and the bearing is measured from broadside

Consider the configuration shown in Fig. 5.6. Here, a line array of N point receiver elements is moving in the $+x$ direction with speed v. The narrow-band source, which is radiating at radian frequency ω_0, is located at x_s and y_s. The signal arriving at the nth element from the narrowband source is given by

$$s_n = a\cos[\omega_0(t - \tau_n(t))], \tag{5.11}$$

where t is the time and τ_n is the time delay associated with the nth element. The crux of the problem lies in the following development leading to $\tau_n(t)$.

When $t = 0$, the range to the first element is taken as a reference and the time delay for the nth element then follows as

$$\tau_n(t) = [R_n(t) - R_1(0)]/c, \tag{5.12}$$

with c being the speed of sound.

$$R_n(t) = \sqrt{(x_s - [(n-1)d + vt])^2 + y_s^2}. \tag{5.13}$$

As mentioned above, the inclusion of the array speed, v, is critical to this development, since it introduces the Doppler into Eq. 5.11 via the time delay. In order to exploit this effect however, the source frequency ω_0 must be known. This means that we must treat the problem as a joint estimation of not only the source coordinates R and θ, but of the source frequency as well. The measurement equations now evolve from the substitution of Eq. 5.13 into Eq. 5.12 and then substituting the result into Eq. 5.11. This is outlined in the following.

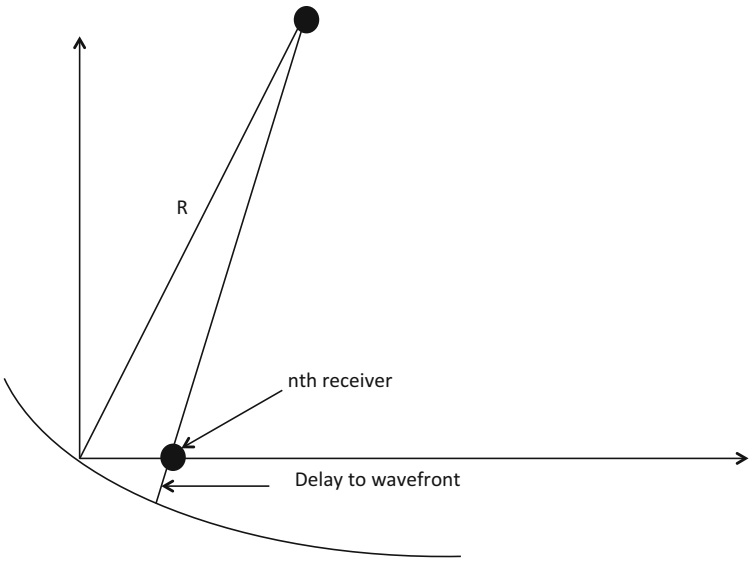

Fig. 5.7 Delay between element and wavefront for wavefront curvature ranging

A vector of the ranges from the source to each element is defined by

$$R(n) = \sqrt{[(x_s - (n-1)d + vt))^2 + y_s^2]}, \quad n = 1, 2, \ldots N. \tag{5.14}$$

A vector of the time delays (see Fig. 5.7) is then defined as

$$\tau(n) = (R(n) - R_{\text{ref}})/c, \tag{5.15}$$

with $R_{\text{ref}} = R(1)$. Here c is the speed of sound. The vector of hydrophone measurements then follows as

$$y(n) = a\cos(2\pi\omega_0(t - \tau(n)). \tag{5.16}$$

This constitutes a measurement vector of the N direct hydrophone measurements. The observed frequency at the first hydrophone is added as an added measurement, making the length of the measurement vector equal to $N + 1$. So that

$$y(N+1) = \omega_0(1 + (v/c)\sin\theta_t), \tag{5.17}$$

with θ_t being the true bearing of the source at the first element. Thus, the nonlinear $N \times 1$ measurement equation system is written more concisely as

$$Y(t) = c[X(t)]. \tag{5.18}$$

The state vector is now given by $X = [x_s \;\; y_s \;\; \omega_0]$, so that

$$X(t|t-1) = AX(t-1|t-1), \tag{5.19}$$

where $A = I$, the identity matrix.

Note that the motion of the source has not been included as it would have been in the usual bearings-only tracker. This is because the Kalman filter is a recursive processor, thereby providing the polar coordinates of the source as a function of time. Indeed, it would be a mistake to include $\dot{x}_s$ and $\dot{y}_s$ as state variables, since this processor depends on the source frequency as a state variable any source motion would play a role in the value of this frequency. Since the Kalman filter can only estimate the *apparent* source frequency, the time varying estimates of the source coordinates themselves constitute the source tracking solution.

Having the state equations and measurement equations, the UKF can now be configured.

5.2.3.1 Example of Joint Bearing and Range Estimation

In this example a 40 element line array, with one-half wavelength element spacing, is towed at a speed of 4 m/s across the zero bearing point of a stationary target, giving an estimate of the time-dependent bearing θ and range[6] R. The UKF configuration follows from Eqs. 5.18 and 5.19. Although we seek the range and bearing, the state actual vector is given by (see Eqs. 5.13–5.17)

$$X = [x_s \;\; y_s \;\; \omega_0]. \tag{5.20}$$

In this way, all of the nonlinearities reside in the measurement system. Two minutes of synthetic data with a frequency of 300 Hz were generated with a -10 dB signal-to-noise ratio at the hydrophone level. At this frequency, with half-wavelength spacing, the length of the array is 97.5 m. The initial values of x_s and y_s used were 300 m and 2,000 m, respectively. The tuning parameters used are

$$P0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 10^{-3} \end{bmatrix} \quad R_{ww} = \begin{bmatrix} 2\times 10^{-4} & 0 & 0 \\ 0 & 4\times 10^{-3} & 0 \\ 0 & 0 & 2\times 10^{-8} \end{bmatrix} \quad R_{vv} = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.25 \end{bmatrix}.$$

The initial state vector is $x0 = [300 \;\; 1500 \;\; 303]'$, and the state transition matrix is the identity matrix. The results are shown in Figs. 5.8 and 5.9.

[6]These estimates are time dependent since the reference used is the first element of the array, and the array is moving. Nevertheless, if the source is also moving, its motion can be found from the resulting range and bearing estimates.

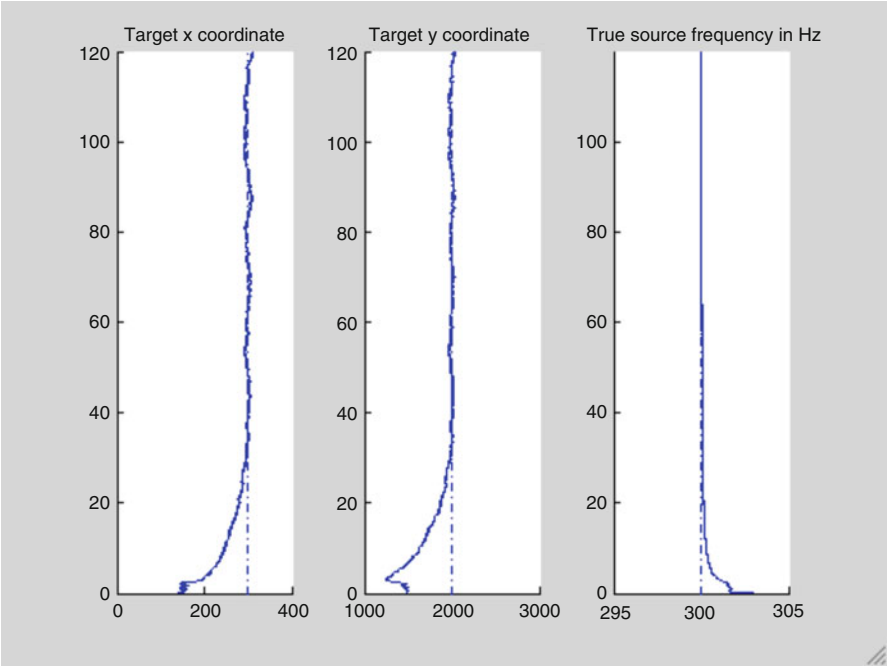

Fig. 5.8 Estimates of state vector elements for a range of about 2,000 m. The *dashed lines* are the true values. The vertical axis is time in seconds and the horizontal axes are in meters

These results offer an opportunity to discuss the importance of the tuning parameters. If we double the range and leave everything else the same, although not shown, the solution requires the full 2 min to converge. However, by quadrupling the value of R_{ww}, the convergence time is significantly reduced, as seen in Figs. 5.10 and 5.11. Note that here, the range is on the order of 40 times the length of the physical aperture.

Finally we show in Fig. 5.12 an example for a range of 10,000 m or 10 km. The tuning parameters are the same except for the fact that R_{ww} was increased by a factor of 10. Here, the only difference is that the convergence time is greater. It is worth noting that the range to physical aperture ratio is 100. We also note that for this case the physical aperture subtends an angle of slightly less than 0.6°. Clearly, this range estimate could not have been achieved using a conventional wavefront curvature approach. At a speed of 4 m/s the array travels 480 m in the 120 s of processing time. Thus somewhat less than five array lengths is traced out for a total baseline of nearly six array lengths. If the model based approach was not used here, this baseline would have been essentially wasted. This algorithm is capable of range estimates far greater than shown here. The limit is really based only on the available SNR, and as one would expect, the greater the range and the noisier the signal, the noisier the estimate and the longer the convergence time.

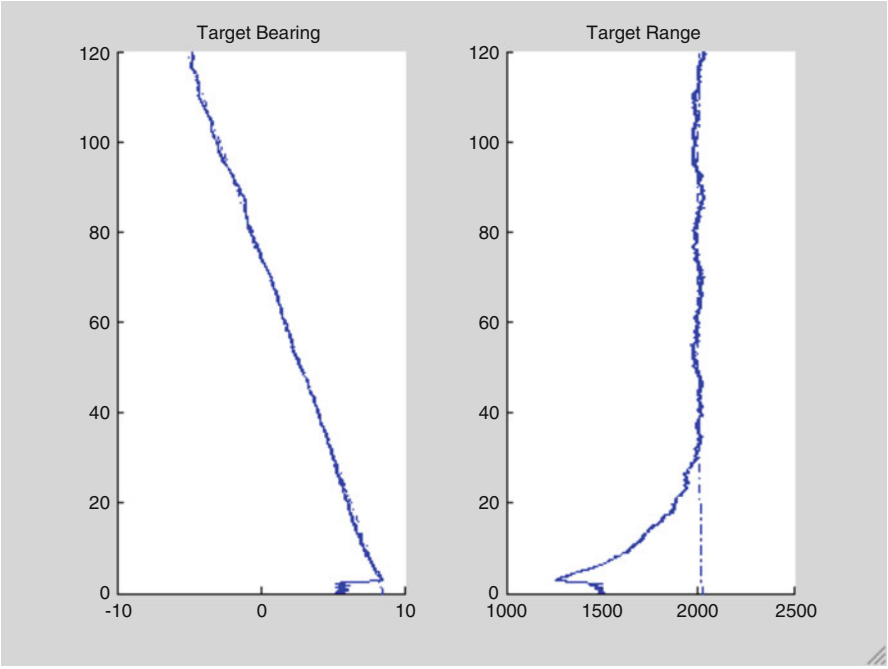

Fig. 5.9 Estimates of the range and bearing for a range of about 2,000 m. The *dashed lines* are the true values. The vertical axis is time in seconds. The horizontal axis on the left-hand plot is in degrees and the horizontal axis on the right-hand plot is in meters

5.3 The Broadband Problem

The issue of the broadband signal requires a quite different approach to the problem. For the narrowband case, the functional form of the signal is explicitly known, where this is not the case for the broadband case. Also, the issue of the Doppler becomes problematical. If we desire to remain in the time domain, rather than exploiting the Doppler, by estimating an apparent source frequency, which would have to be found by using a Fourier transform, we could proceed by using the Doppler equation to estimate a virtual sample rate at the source. This virtual sample rate would then be used to discretize the time delay. However, this presents its own problem, since the sample rate must be high enough to provide the necessary precision in the time delay estimate, resulting in a prohibitive computation time. Nevertheless, this is still a viable approach.

The approach we will take here is to work in the frequency domain. This presents its own problems. First, the data must be preprocessed—that is, the measurement system does not simply consist of the hydrophone measurements. This is a consequence of the fact that an explicit model of the time domain signal does not exist in a closed form. The second complication is that the phases of the hydrophone signals must be obtained via a phase unwrapping procedure, which

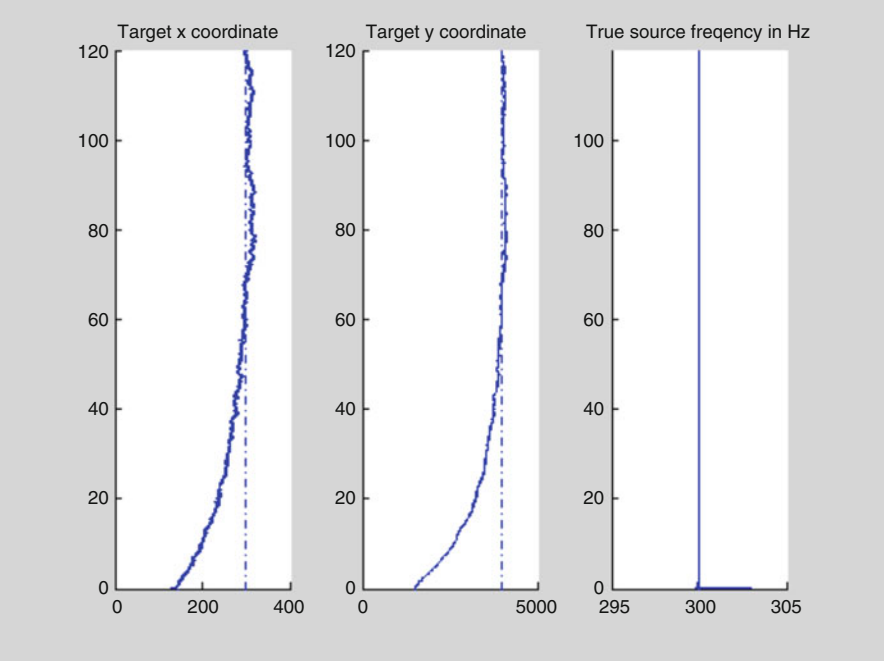

Fig. 5.10 Estimates of state vector elements for a range of about 4,000 m. The *dashed lines* are the true values. The vertical axes are time in seconds. The left and center horizontal axes are in meters and the right-hand horizontal axis is in Hz

poses a limit on the SNR. That is, unlike the previous examples, the results do not "gracefully degrade" as the SNR decreases, since the phase unwrapping process is strongly nonlinear. That is, as the SNR decreases, a point is reached where the phase unwrapping calculation collapses.

5.3.1 Frequency Domain Broadband Array Processor: Theory

Consider first a narrow-band version of the problem. Consider a line array of N receiver elements to be moving with speed v in the $+x$ direction of an $x-y$ coordinate system. Let the plane-wave signal be arriving at angle θ with respect to the y axis, measured to be positive for clockwise rotation. The signal at the nth receiver can then be represented in complex form as

$$s_n(t) = a_n e^{i(\omega_0/c)(nd+vt)\sin\theta + i\omega_0 t}. \tag{5.21}$$

As before, ω_0 is the radian frequency of the signal at the source, d is the coordinate of the nth receiver element, θ is the bearing angle measured from broadside, a_n is the signal amplitude, and t is time. Since we will be working in the phase domain,

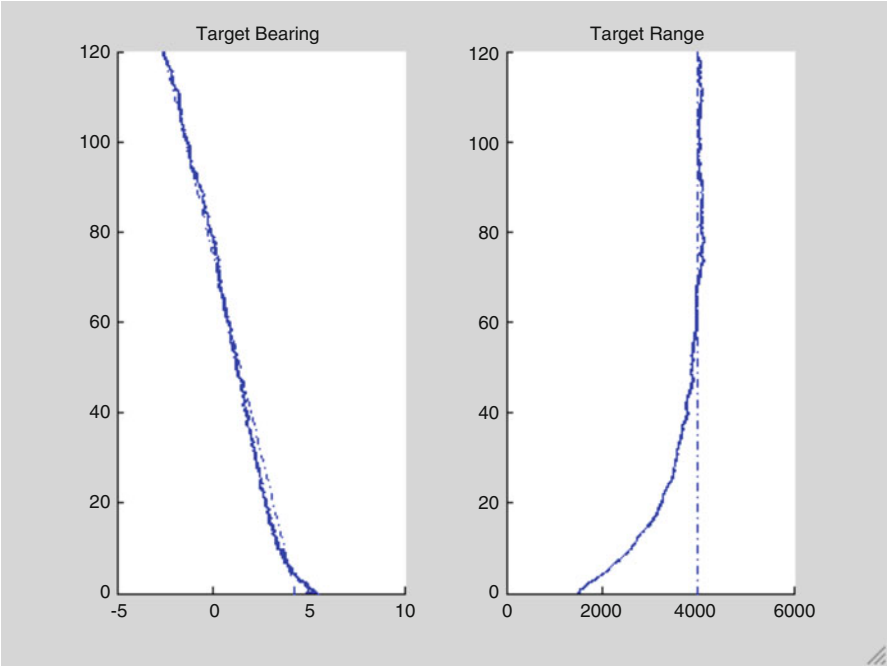

Fig. 5.11 Estimates of the range and bearing for a range of about 4,000 m. The *dashed lines* are the true values. The vertical axis is time in seconds. The left-hand horizontal axis is in degrees and the right-hand horizontal axis is in meters

there is no need to include the signal amplitude as a nuisance parameter. This leaves the two parameters ω_0 and θ to be estimated. The state equation of the Kalman filter is given by

$$\begin{bmatrix} \theta(t|t-1) \\ \omega_0(t|t-1) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \theta(t-1|t-1) \\ \omega_0(t-1|t-1) \end{bmatrix}. \tag{5.22}$$

If there is a non-negligible bearing rate in the scenario, this can be easily dealt with by augmenting a bearing rate into the Kalman equations. For this case, Eq. 5.22 is generalized to

$$\begin{bmatrix} \theta(t|t-1) \\ \alpha(t|t-1) \\ \omega_0(t|t-1) \end{bmatrix} = \begin{bmatrix} 1 & \Delta t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \theta(t-1|t-1) \\ \alpha(t-1|t-1) \\ \omega_0(t-1|t-1) \end{bmatrix}, \tag{5.23}$$

where α is an estimate of $\partial\theta/\partial t$, and Δt is the update time increment.

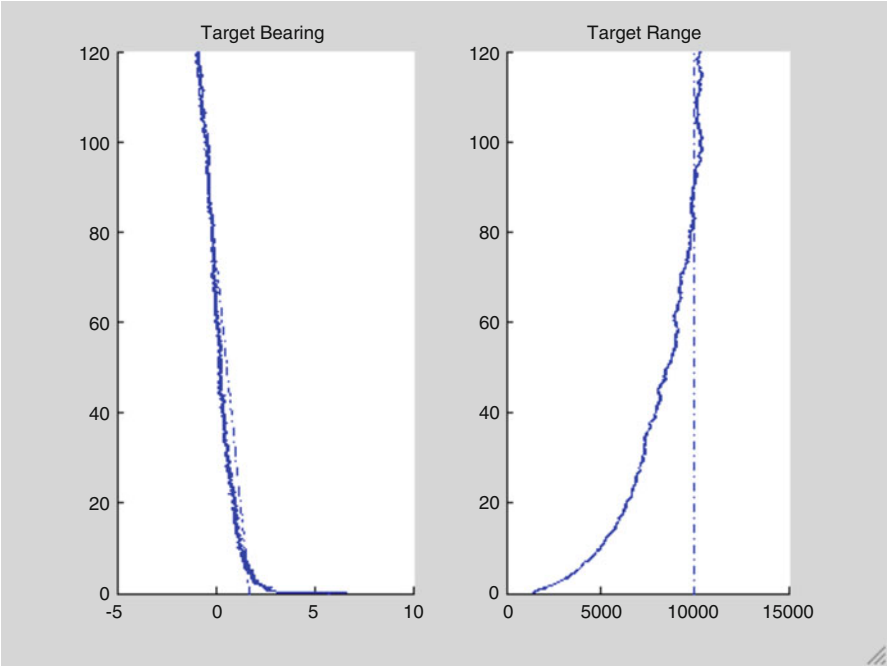

Fig. 5.12 Estimates of the range and bearing for a range of about 10,000 m. The *dashed lines* are the true values

Now consider the broadband case. Since we choose to work in the phase domain, the measurement equation is based on the exponent in Eq. 5.21. This means that we must preprocess each hydrophone signal by creating a sequence of short time segments which will be transformed into the frequency domain. A discrete spectrum of the broadband signal for each of these time segments is computed. Consider now one of these segments for the nth hydrophone. Each frequency will have a phase associated with it. Since these phases are related by the frequency index, an average phase for this time segment can be computed as

$$\begin{aligned}\psi_n &= \frac{1}{\Delta m}\sum_{m=M_{\mathrm{Lo}}}^{M_{\mathrm{Hi}}}\{m(\omega_0/c)(nd+vt)\sin\theta + m\omega_0 t + \phi_{mn}\}/m \\ &= \frac{1}{\Delta m}\sum_{m=M_{\mathrm{Lo}}}^{M_{\mathrm{Hi}}}\{m(\omega_0/c)nd\sin\theta + \phi_{mn}\}/m + \omega t.\end{aligned}$$

Here, $\omega = \omega_0(1 + (v/c)\sin\theta)$ and is the lowest frequency of a DFT of the data at the receiver and M_{Lo} and M_{Hi} are the respective low frequency and high frequency indices of this DFT. $\Delta m = M_{\mathrm{Hi}} - M_{\mathrm{Lo}} + 1$ is the number of frequency components of the DFT, $d = x_n - x_{n-1}$ is the inter-element spacing of the line array receiver

elements, ϕ_{mn} is an arbitrary phase, and m is the frequency index. Note that this refers all of the phases to that of the lowest frequency line of a virtual DFT at the source.

Since only the phase *differences* are relevant to the problem, there are only $N-1$ receiver-based phase difference measurements available for the N receivers. Assuming that the arbitrary phase terms ϕ_{mn} average to a negligibly small value, the $N-1$ measurements, based on the above model for the phase are given by

$$y_n = \psi_{n+1} - \psi_n, \qquad n = 1, 2, \ldots, N-1, \tag{5.24}$$

where the ωt term has canceled out. There is an auxiliary measurement equation that is based on the observed frequency. This is basically the Doppler relation and is given by

$$y_N = \omega = \omega_0(1 + (v/c)\sin\theta). \tag{5.25}$$

In this equation, ω_0 can be thought of as the lowest frequency of a virtual DFT of the signal at the source. The resulting measurement system has the (nonlinear) form

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N-1} \\ y_N \end{bmatrix} = \begin{bmatrix} (d/c)\omega_0\sin\theta \\ (d/c)\omega_0\sin\theta \\ \vdots \\ (d/c)\omega_0\sin\theta \\ \omega_0 + (v/c)\omega_0\sin\theta \end{bmatrix}, \tag{5.26}$$

or symbolically,

$$Y = c[\theta, \omega_0]. \tag{5.27}$$

5.3.2 The Algorithm

As mentioned previously, the hydrophone data must be preprocessed before arriving at the above measurement equations. Referring to Fig. 5.13 the hydrophone output time series is segmented into 0.1 s segments,[7] bandpass filtered to the desired section of the spectrum, decimated to a manageable sample rate and then put through an FFT. The phase of each hydrophone signal is then found by dividing out the index of each frequency line and then averaging. That is, by dividing out the frequency index, the frequencies, and thereby the phases, are aligned with the lowest value ω_0, and

[7] This value is that selected for the experimental example which follows in the next section.

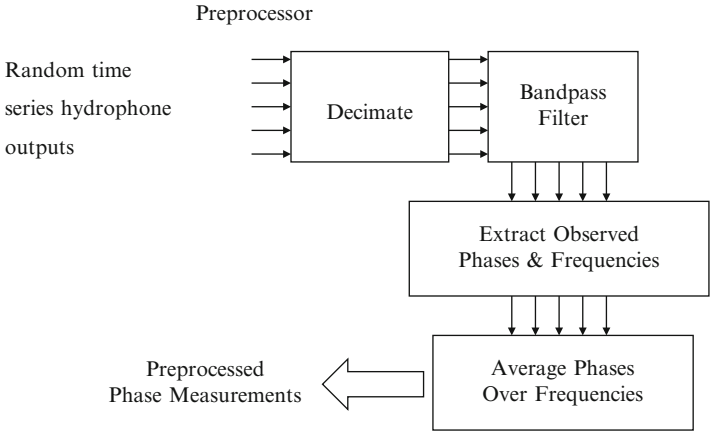

Fig. 5.13 Flow diagram for the broadband data preprocessor

thereby can be averaged to provide an estimate of phase. This averaging procedure includes a phase unwrapping step. This is the weak point of this processor, since for low SNR, the phase unwrapping step fails.

5.3.3 Experimental Results

The work presented here was originally reported in [16]. The experimental data were obtained as a data set of opportunity. During an experiment carried out jointly by Boston University and Woods Hole Oceanographic Institution, using the autonomous Undersea vehicle REMUS, a short (six element) array was towed. During the experiment, a ferry from the mainland of Cape Cod on its way to the island of Nantucket passed through the area. The resulting data provide the basis for this work [10].

The six element array, which had an element spacing of 0.75 m, was towed at a speed of 1.5 m/s. The ferry appeared by emerging from a shallow region, known as Tuckernuck Shoal, at an angle very close to broadside (0°) to the towed array, and the closest point of approach occurred at approximately 20°. The array was moving in a straight line toward the course of the ferry, which was moving at approximately 20 kts, on a straight course from left to right with respect to forward endfire of the array. This configuration is depicted in Fig. 5.14 where points A and B are the ferry positions for the respective beginning and CPA of the data used in this work. The distance between these two points is approximately 2 km. Although the radiated sound from the ferry was quite broadband, extending over a band from about 100 to 1,000 Hz, there was a particularly strong band of energy occurring between 890 Hz and 920 Hz. This energy band was selected for the bearing estimation. At this band of frequencies, the array has an acoustic length of approximately 2.3 wavelengths.

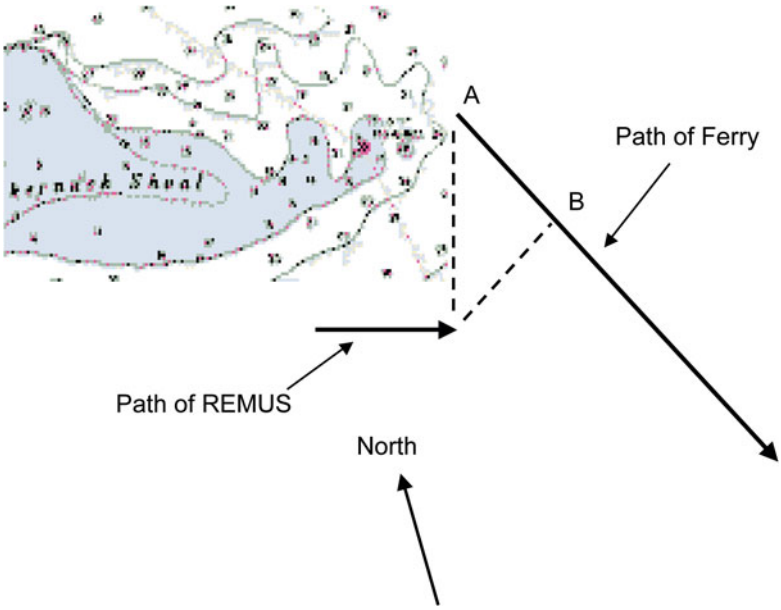

Fig. 5.14 Configuration of the experiment

A sequence of 8000 0.1 s DFTs was generated in order to obtain the phase averages over the full 800 s of the data. The band was then constrained to the lines between 890 Hz and 920 Hz. The phase averages extracted from each hydrophone sequence constituted the basis of the measurements used in Eq. 5.26. The frequency measurement was taken directly as the lowest frequency of the DFT of the data.

The results are shown in Figs. 5.15 and 5.16. In both figures the vertical axis is time in seconds. The left panel of Fig. 5.15 is the result of beamforming the data with a conventional frequency-domain beamformer, and is computed from

$$p(t_j, \theta_k) = \sum_{n=1}^{N} \sum_{m=M_{\mathrm{Lo}}}^{M_{\mathrm{Hi}}} \left\{ e^{-i2\pi f_m n (d/c) \sin\theta_k} \right\} F_{t_j}^*(f_m, n). \tag{5.28}$$

The bracketed term is the steering vector and $F_{t_j}(f_m, n)$ is the frequency domain signal at the nth hydrophone associated with the time t_j. The center panel shows the maxima of the plot in the left panel, and the right panel shows the model-based result. As expected, both estimators fail to resolve the bearing in the neighborhood of endfire. After endfire, beginning at about 400 s, the model-based processor clearly shows the cumulative performance improvement expected of such a processor. This occurs since the bearing rate is small, and therefore the random walk state model of Eq. 5.22 is valid.

Figure 5.16 depicts the results for the case where the bearing rate is augmented into the processor. The left panel shows the bearing estimate, the center panel shows the estimate of the bearing rate, and the right-hand panel shows the estimate of

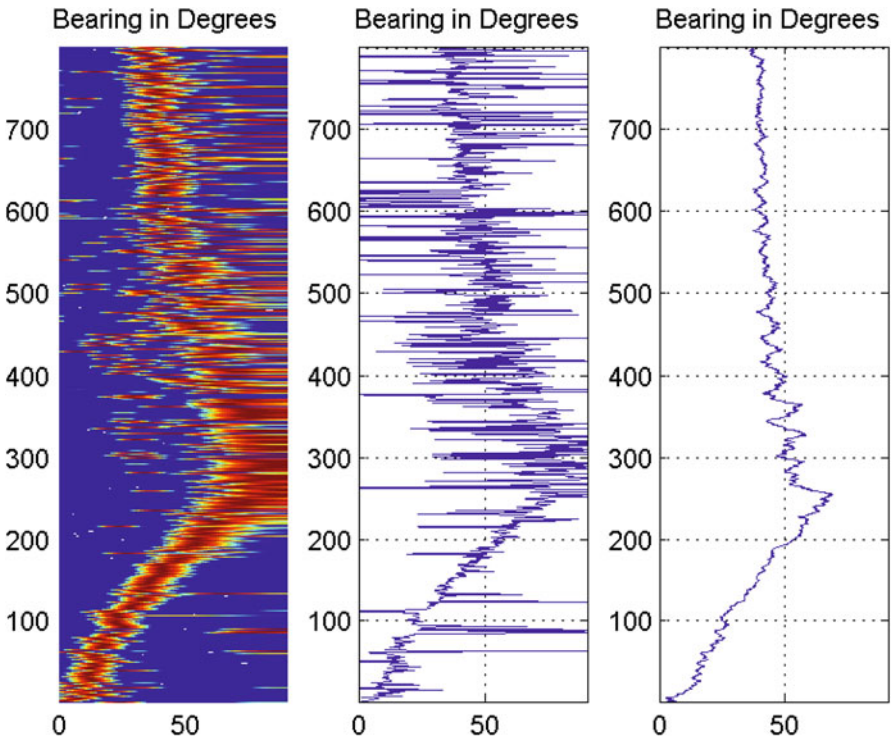

Fig. 5.15 Bearing estimation results. The *left panel* is the conventional frequency-domain beamformer result. The *middle panel* is a plot of the maxima of the plot in the *left panel* and the *right panel* is the model-based processor result

the source fundamental frequency. Note that this frequency is not constant, since the source itself is undergoing nonzero accelerations. Thus, before endfire it has an up Doppler and after endfire, a down Doppler. Thus, the apparent fundamental frequency of the virtual DFT at the source must adapt to these speed changes.

The fact that the bearing estimate in Fig. 5.16 shows some improvement over that of Fig. 5.15 is a direct consequence of the inclusion of the bearing rate as an augmented state vector element. The Kalman filter requires that the user specify a trial value for the state error covariance. The value chosen constitutes a lower bound on the eventual state error covariance. This provides a means for the user to control the convergence rate of the process. That is, the larger this covariance is chosen to be, the faster the convergence of the processor; but at the price of a noisier estimate. The estimate in Fig. 5.16 allowed a smaller value for this covariance to be used, since the convergence requirements for the case of a nonzero bearing rate are eased by the inclusion of the bearing rate directly into the dynamics via Eq. 5.23. Thus, the limiting state estimation error is smaller in Fig. 5.16 (left panel), than that in Fig. 5.15 (right panel).

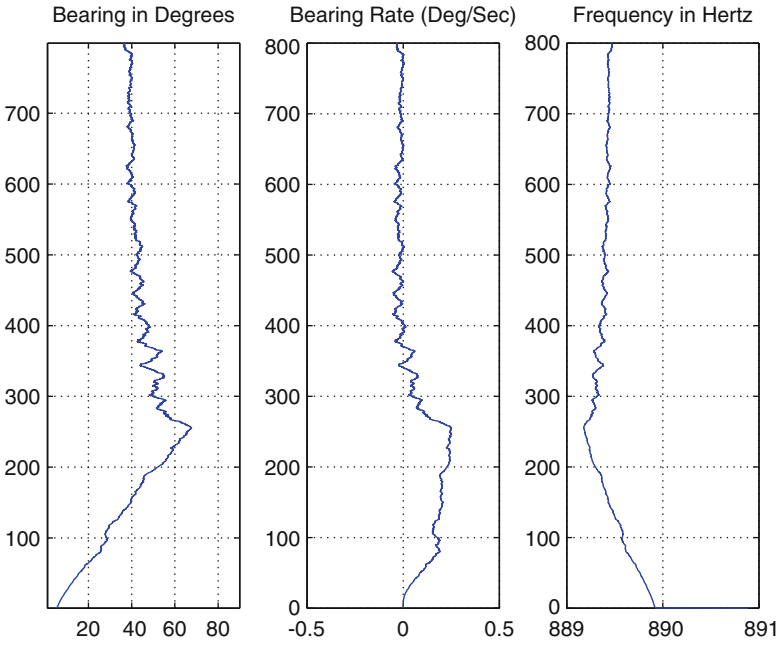

Fig. 5.16 Bearing rate augmentation result

5.4 Model-Based Localization

In the first chapter, the matched-field problem (MFP) was discussed in terms of its limitations, in particular, the *mismatch problem* where lack of accurate knowledge of the model parameters either degrades the solution or indeed, prevents any viable solution at all. An example [9] was shown in which an MFP solution using experimental data which supported nine modes could not be achieved without eliminating the top two modes, i.e., modes eight and nine. This means that these two modes provided by the model were simply not an accurate representation of their actual counterparts. In 1995, Candy [5] applied the concept of augmentation to this type of problem and demonstrated that in this way, the modal functions themselves could be jointly estimated along with the modal amplitudes, and those modes provided by the propagation model served as the initial values provided to the Kalman filter. It is worth outlining this approach since it not only demonstrates the power of augmentation, but it also provides an example of the use of the innovations sequence as a measure of the fidelity of the model. In the example to follow, there were 23 hydrophones in the vertical array and 235[8] state vector

[8]The five modal functions and their derivatives were each evaluated at 23 points along with the five modal amplitudes.

elements. This configuration allows the hydrophone measurements to be treated as a spatial series of measurements whereby the recursive nature of the processor can treat them with a single measurement equation rather than one with a dimension of 23×1. This provides a significant increase in computational efficiency. The following example is based on a set of data obtained by Carey et al. [6] in the Hudson Canyon area off of the New Jersey coast. The depth was 72 m. A 23 element vertical array with 2.5 m spacing was used, thereby spanning 55 m of depth.

We first review how this problem would be formulated as an MFP. The Helmholtz equation is put into cylindrical coordinates denoted by r ϕ and z. The equation is separable in this coordinate system. Here r is the radial coordinate, which for our problem is the range, z is the depth coordinate and the ϕ coordinate is ignored, since the problem is conventionally viewed in two dimensions. This is consistent with Fig. 1.2. Upon separation, the range equation is of the form of Bessel's equation, and for the case of propagating waves, the relevant solutions are zeroth order Hankel functions of the first kind, designated by $H_0^1(k_r(m)r)$ for outward going waves. Here $k_r(m)$ is the radial wavenumber associated with the mth eigensolution, The equation in z is an eigenvalue equation given by

$$\frac{d^2}{dz^2}\phi_m(z) + k_z(m)\phi_m(z) = 0; \qquad m = 1, \ldots, M, \tag{5.29}$$

where there are M eigensolutions and the wavenumbers, by virtue of the separation process, are related by the so-called *dispersion relation*, viz.

$$k^2 = k_z^2(m) + k_r^2(m), \tag{5.30}$$

with $k = \omega/c(z)$ being the water wavenumber. Here $\omega = 2\pi f$, f is the frequency, and $c(z)$ is the speed of sound in the water, which depends upon the depth. Thus, each modal function $\phi_m(z)$ is associated with a range solution $H_0^2(k_r(m)r)$ by virtue of the dispersion relation. This is a range-independent form of the wave equation, meaning that there is no dependence of the sound speed on r. Having these solutions in hand, the total solution is then given by

$$p(r_s, z) = q \sum_{m=1}^{M} H_0^1(k_r(m)r_s)\phi_m(z_s)\phi_m(z), \tag{5.31}$$

where p is the acoustic pressure, q is the source amplitude, ϕ_m is the mth modal function at z and z_s is the source depth, $k_r(m)$ is the horizontal wavenumber associated with the mth mode, and r_s is the source range. It is useful to write this solution in the form

$$p(r_s, z) = \sum_{m=1}^{M} \beta_m(r_s, z_s)\phi_m(z), \tag{5.32}$$

where $\beta_m(r_s, z_s) = qH_0^1(k_r(m)r_s)\phi_m(z_s)$ is called the mth modal amplitude.

The usual MFP would now proceed as follows. The vertical array provides a vector of measurements of the true field, which we call P_t. The model provides a prediction of these measurements for a given set of source coordinates, z_s r_s. This we label as P_p. The solution, as first put forth by Bucker [2], is found by seeking the maximum of the magnitude squared of the normalized inner product of these two vectors. That is, the estimates of r_s and z_s are found from

$$\text{Max}\, A(r,z) = A(r_s,z_s) = \text{Max}\, \frac{|P_t' P_p|^2}{|P_t|^2 |P_p|^2}. \tag{5.33}$$

This correlation surface (sometimes referred to as the "ambiguity" surface[9] approach is not optimal, since it is neither a maximum likelihood or minimum variance solution [14], even in the case of a perfectly known set of modal functions. The maximum is found by exhaustively searching over the surface $A(r,z)$. In practice however, as mentioned in Chap. 1, unless the model parameters, in this case, the modal functions, are known to a sufficient degree of accuracy, a viable solution cannot be found. Much effort has been expended trying to alleviate this problem, leading to improvement in some cases. However, as can be seen from Eq. 5.32 the obvious approach is to jointly estimate the modal functions and their amplitudes jointly, since using the accuracy of the modal functions obtained from a model are subject to the uncertainties in the sound speed profile (SSP) and boundary conditions at the bottom, which are themselves open to inaccuracies.

A better approach is to carry out the solution in modal space. This is done by observing that Eq. 5.32 is linear in the modal functions, so that if the number of measurements equals or exceeds the number of modes, the modal amplitudes can be found by using a pseudo inverse [15], sometimes called the *Moore–Penrose* inverse, which leads to a maximum likelihood solution for the modal amplitudes. This approach offers some advantage *vis a vis* the mismatch problem, since if the troublesome modes can be identified, they can simply be removed. However, this entails the cost of reducing the information content of the model.

Returning now to Candy's approach, we begin by observing that following Eqs. 3.89 and 3.90 from Chap. 3, Eq. 5.29 can be written in state space as follows. Since each mode is characterized by a second order, ordinary differential equation, its state space form is

$$\frac{d}{dz}\underline{\phi}_m(z) = A_m(z)\underline{\phi}_m(z), \tag{5.34}$$

with

$$A_m(z) = \begin{bmatrix} 0 & 1 \\ -k_z^2(m) & 0 \end{bmatrix}, \tag{5.35}$$

[9]This terminology, although commonplace, is unfortunate, since it is not the ambiguity diagram as originally defined by Woodward [17] which plays an important role in the active sonar problem.

and

$$\underline{\phi}_m(z) = [\phi_m(z) \;\; \dot{\phi}_m(z)]' \tag{5.36}$$

with $\dot{\phi}_m(z) = \frac{d}{dz}\phi_m(z)$. Including all M modes leads to

$$\frac{d}{dz}\underline{\phi}(z) = A(z)\underline{\phi}(z), \tag{5.37}$$

where $\phi_m(z)$ the $2M \times 1$ state vector defined by

$$\underline{\phi}(z) = [\phi_1(z) \; \dot{\phi}_1(z) \;\; \phi_2(z) \; \dot{\phi}_2(z) \; \cdots \; \phi_M(z) \; \dot{\phi}_M(z)]', \tag{5.38}$$

and the $2M \times 2M$ block diagonal transition matrix is given by

$$A(z) = \begin{bmatrix} A_1(z) & & \\ & \ddots & \\ & & A_M(z) \end{bmatrix}. \tag{5.39}$$

The state space propagator now obtains from this by discretization, i.e.,

$$\underline{\phi}(z_\ell) = [I + \Delta z A]\underline{\phi}(z_\ell), \tag{5.40}$$

where ℓ is the hydrophone index. Defining $I + \Delta z A$ as $A(z_{\ell-1}, \beta)$, and assuming a random walk model for the modal amplitudes $\{\beta_i\}$ with

$$\underline{\beta} = [\beta_1 \; \beta_2 \; \cdots \; \beta_M]', \tag{5.41}$$

leads to the *augmented* state space model given by

$$\begin{bmatrix} \underline{\phi}(z_\ell) \\ --- \\ \underline{\beta}(z_\ell) \end{bmatrix} = \begin{bmatrix} A(z_{\ell-1}, \beta) & | & 0 \\ - & - & - \\ 0 & | & I_M \end{bmatrix} \begin{bmatrix} \underline{\phi}(z_{\ell-1}) \\ --- \\ \underline{\beta}(z_{\ell-1}) \end{bmatrix} + \begin{bmatrix} \underline{w}(z_{\ell-1}) \\ --- \\ \underline{w}_\beta(z_{\ell-1}) \end{bmatrix}, \tag{5.42}$$

with the associated nonlinear measurement model which follows directly from Eq. 5.32

$$p(r_s, z_\ell) = c[\underline{\beta}(z_\ell), \underline{\phi}(z_\ell)] + v(z_\ell). \tag{5.43}$$

This is a Gauss–Markov representation which includes the second order statistics. The measurement noise can represent the near-field acoustic noise field, flow noise on the hydrophone, and electronic noise. The modal (process) noise can represent sound speed errors, distant shipping noise, errors in the boundary conditions, sea state effects, and ocean inhomogeneities.

Note that Eq. 5.43 contains the hydrophone measurements as a series, much analogous to a time series. This means that the measurement equation can be treated as a single equation which treats the hydrophone data sequentially. As mentioned before, this provides a huge saving in computation time. Also, in the results to follow, a single pass over the array was sufficient to provide a viable solution. This means that each of the five modal functions (and their derivatives) supported by the propagation channel in the following problem will be estimated at 23 points, along with the five estimates of the modal amplitudes. This points out the fact that even though there are more estimates than measurements, the estimates of the 23 points of each of modal functions are not independent. This is a consequence of the fact that the transition matrix that evolved from the vertical equation contains the information contained in the vertical equation.

For the solution to proceed, the horizontal wavenumbers are required. These were found from the experimental data [4] by computing its spatial spectrum, in which the spectral lines are the horizontal wave numbers. The solution is found in two steps. First, Eqs. 5.42 and 5.43 are solved as a Kalman filter problem, leading to estimates of the modal functions evaluated at 23 points, and the five modal amplitudes. The state vector was initialized by using solutions for the modal functions and the modal amplitudes from the SNAP [11] model. The resulting modal amplitudes are implicit functions of the source range and depth, r_s and z_s. Thus a second step is required which is a numerical optimization carried out by finding the minimum of

$$J(r_s, z_s) = \sum_{m=1}^{M} [\beta(r_s, z_s) - H_0^1(k_r(m) r_s)\phi_m(z_s)]^2. \tag{5.44}$$

There two issues complicating this calculation. First, the surface $J(r_s, z_s)$ is not unimodal. In [5], in order to find the global minimum, the polytope method of Nelder–Meade [12] was used. This method tests each extremum found by testing in its neighborhood. Alternatively, the surface could be exhaustively searched as is done in the conventional MFP problem. This is inefficient since it requires the complete range-depth map to be computed over the region of interest. In this work, the Nelder–Meade method was used.

The second issue is that unless there is some means of interpolating the modal function estimates is used, the depth estimate accuracy will be limited to the precision allowed by the 23 point estimate. This could be done by one of two ways. Either the modal function points could be fit to a curve or the actual propagator could be used as a way of generating smaller steps, which was done here. For more detail on this, see [5]. As previously mentioned, the solution was carried out using only a single pass over the array. The resulting estimates of the modal functions are shown in Fig. 5.17. The red lines are the two sigma bounds on the mean. The solution for the source coordinates found from the Nelder–Meade optimizer are plotted in Fig. 5.18.

As can be seen, this solution is remarkably accurate. This is mainly due to the fact that the modal functions are estimated directly from the data. Unlike MFP,

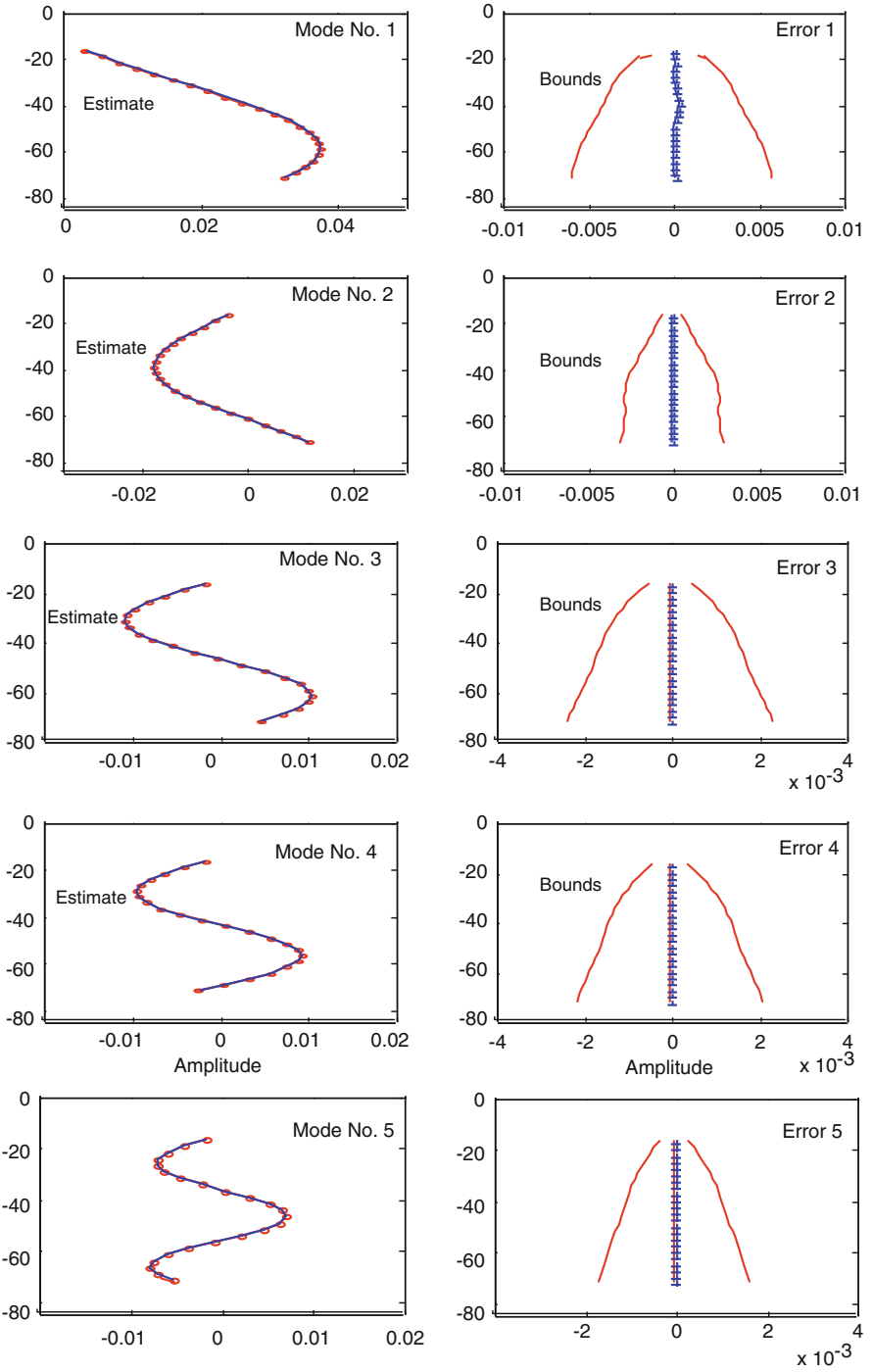

Fig. 5.17 Estimates of the modal functions

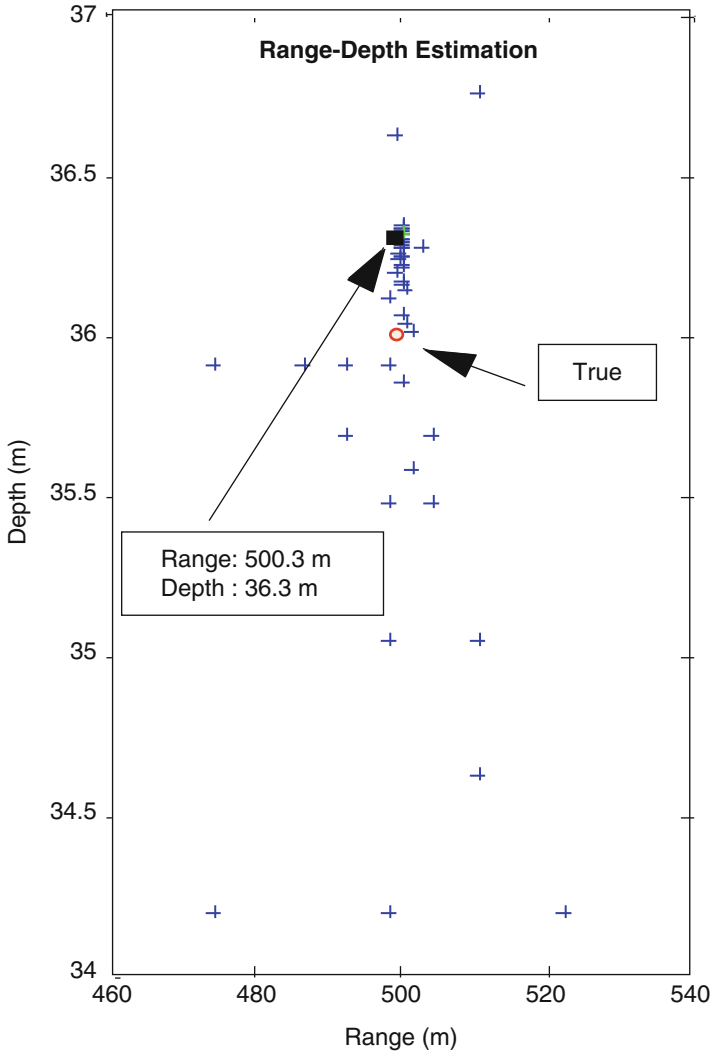

Fig. 5.18 Localization solution using the Nelder–Meade optimizer

the vertical equation is imbedded directly into the state equation, whereas in MFP, the modal functions are predicted from the model, which in turn requires accurate knowledge of the boundary conditions, especially those at the bottom, and an accurate knowledge of the SSP. In the model-based approach, these predicted modal functions are used to initialize the state vector, but they are then adaptively corrected by the data vis the recursive estimation algorithm. At this point, any errors in the boundary conditions and the SSP become moot, since the data itself is now the determining factor.

For a complete description of the solution, the reader is directed to [5].

References

1. Aidala, V., Hamel, S.: Utilization of modified polar coordinates for bearings-only tracking. IEEE Trans. Autom. Control **AC-28** (1983)
2. Bucker, H.P.: Use of calculated sound fields and matched-field detection to locate sound sources in shallow water. J. Acoust. Soc. Am. **59**, 368–372 (1976)
3. Candy, J.V.: Bayesian Signal Processing: Classical, Modern, and Particle Filtering Methods. Wiley, New York (2009)
4. Candy, J.V., Sullivan, E.J.: Model-based processor design for a shallow water ocean acoustic experiment. J. Acoust. Soc. Am. **95**, 2036–2051 (1993)
5. Candy, J.V., Sullivan, E.J.: Passive localization in ocean acoustics: a model-based approach. J. Acoust. Soc. Am. **96**, 1455–1471 (1995)
6. Carey, W., Doutt, J., Evans, R., Dillman, L.: Shallow water transmission measurements taken on the New Jersey continental shelf. IEEE J. Oceanic Eng. **20** (4) 321–336 (1995)
7. Frankel, T.: The Lie Geometry of Physics, pp 125–154. Cambridge University Press, Cambridge (2011)
8. Hedrick, J.K., Girard A.: Controllability and observability of nonlinear systems. http://www.me.berkeley.edu/ME237/6_cont_obs.pdf (2005)
9. Hinich, M.J., Sullivan, E.J.: Maximum-likelihood passive localization using mode filtering. J. Acoust. Soc. Am. **85**, 2214–219 (1989)
10. Holmes, J.D., et al.: An autonomous underwater vehicle towed array for ocean acoustic measurements and inversions. In: Proceedings of IEEE Oceans 2005 - Europe, 20–23 June, vol. 2, pp. 1068–1061 (2005)
11. Jensen, F.B., Ferla, M.C.: SNAP: the SACLANTCEN normal-mode acoustic propagation model. SACLANTCEN memorandum NATO (1979)
12. Nelder, J.A., Meade, R.: A simplex method for function minimization. Comput. J. **7**, 308–313 (1965)
13. Sullivan, E.J., Candy J.V.: Space-time array processing: the model-based approach. J. Acoust. Soc. Am. **102**(5), 2809–2820 (1997)
14. Sullivan, E.J., Middleton, D.: Estimation and detection issues in matched-field processing. IEEE Trans. Ocean. Eng. **18**, 321–336 (1995)
15. Sullivan, E.J., Candy, J.V., Persson L.: Model-based acoustic array processing. In: Proceedings of the 1st International Conference on Underwater Acoustic Measurements. Also in Lawrence Livermore Report UCRL-CONF-210525. March 14 2005
16. Sullivan, E.J., et al.: Broadband synthetic aperture: experimental results. J. Acoust. Soc. Am. **120**, EL49 (2006)
17. Woodward, P.M.: Probability and Information Theory with Applications to Radar. Artech House, Norwood (1980)

Chapter 6
Filter Tuning and Solution Testing

6.1 Discussion

Since this is a monograph, there are necessarily subjects that have not been fully covered. Here we wish to amend this to some degree. First, although the science behind the modeling and the mathematics of the Kalman filter have been addressed, there are two issues that, until now, have been neglected. These are the process of "tuning" the Kalman filter, and that of assessing the quality of the solutions. The solution quality is mainly based on testing of the innovations sequence. The innovations sequence is, in essence, a continual observer of the progress of the Kalman filter, since it sequentially compares each new measurement to what the model thinks it will be. As such it carries highly useful information on how the processor can be improved. As a simple example, at the end of the last chapter the model-based localization problem was presented. Here we recall that according to the data, there were five modes. This was known a priori since it was determined by a spatial spectrum of the data taken during the experiment. However, suppose this were not the case. That is, suppose that it were assumed that there were only four modes. In such a case, the spectrum of the innovations sequence would have indicated this by presenting the spatial spectral line of the fifth mode, since it was contained in the data. Speaking more generally, the innovations not only indicated modeling errors, but in many cases it can actually identify the problem.

6.2 Tuning the Filter

As previously discussed, the Kalman filter inputs R_{ww}, R_{vv}, and $P0$ must be selected by the user. These are called the tuning parameters, and selecting them is considered to be an art more than a procedure that can be codified in a few steps. In order to understand this it is useful to look at the Kalman gain term K, where

E.J. Sullivan, *Model-Based Processing for Underwater Acoustic Arrays*,
SpringerBriefs in Physics, DOI 10.1007/978-3-319-17557-7_6

$$K = \tilde{P}C'R_{\epsilon\epsilon}^{-1}, \tag{6.1}$$

and examine its role in the update step of the algorithm. In the following simplified expression, X_c, the corrected or updated state vector is found by adding the correction term as follows.

$$X_c = X_p + K\epsilon, \tag{6.2}$$

with X_p being the predicted state vector. If the state error covariance $\tilde{P}$ is large, then this acts to make the Kalman gain large, indicating that the filter is providing a noisy estimate and therefore the role of the measurements, via the innovation, becomes more important than the model. That is, the algorithm believes more strongly in the data than in the model. Of course, the opposite is true if the state error covariance is small, rendering the value of K to be small. In this case the model takes more precedence *viv a vis* the measurements. A similar argument holds with regard to the innovations covariance $R_{\epsilon\epsilon}$. Since it enters the gain in terms of its inverse, if it is large, so that its inverse is small, this means that the measurement noise is large. This acts to make the Kalman gain term small, meaning that the filter depends more on the model. Conversely, it then follows that if the measurement noise is small, then this tends to increase the Kalman gain, thereby putting more dependence on the measurements.

The above argument provides a guide for the selection of the values of R_{ww} R_{vv}. Ideally, the value of R_{vv} should match the true measurement noise covariance, but this is not always known to a sufficient accuracy, so intuition gained from experience must become the guide. Clearly, if R_{vv} is too large, it will underemphasize the role of the measurements, putting too much dependence on the model, whereas the opposite, of course, is true, i.e., if it is unrealistically small, then the measurements will be overemphasized relative to the model.

Also, by observing that the value of $\tilde{P}$ can never be less than R_{ww}, it is important to have this term as small as practically possible so as to aim for a small state error. However, if it is too low, the speed of convergence of the filter becomes unacceptably slow. Furthermore, since the role of R_{ww} is to compensate for modeling errors, making it too small can put too much focus on the deficiencies of the model. Thus, it can be seen that experience must play a role here also. Generally speaking, the larger R_{ww}, the faster the convergence, but the noisier the estimates, and the smaller the value of R_{ww}, the noise on the state estimate is reduced, but at the cost of slower convergence.

The selection of $P0$ also deserves some attention, since selecting it to be too small or too large will delay convergence. Also if $P0$ is selected to be too large, depending on the model, it can cause the filter to deviate from a convergent path, thereby preventing any solution.

6.3 Assessing Solution Quality

There are two tests that can be made to assess the quality of the solution. First, the state error bounds can be compared to the state estimate. The red curves in the right column of Fig. 5.17 are the two sigma bounds for the modal estimates and are found directly from the associated component of the diagonal of the state error covariance. In these plots, the means of the initial modes has been subtracted from the means of the estimated modes. This test evaluates the enhancement obtained by using the physical model in the estimation scheme.

6.3.1 Innovation Sequence Zero Mean Test

Of no less importance are the tests one makes on the innovations. If the innovation sequence is zero mean and white (uncorrelated), then the solution is considered to be optimum [1]. The zero mean test is based on the sample mean of the innovations sequence. This sample mean, for the ith component of $\epsilon_i(t)$, is given by

$$\hat{m}_\epsilon(i) = \frac{1}{N}\sum_{i=1}^{N} \epsilon_i, \tag{6.3}$$

and for N reasonably large, $\hat{m}_\epsilon(i) \sim \mathcal{N}(m_\epsilon, \sigma^2_{\epsilon\epsilon}/N)$ where N is the number of samples. For a normalized Gaussian the 5 % significance level is given by α, which is defined by

$$\frac{1}{\sqrt{2\pi}}\int_{-\alpha}^{\alpha} e^{-x^2/2}dx = 0.95. \tag{6.4}$$

The resulting value of α is 1.96. If we now define $x = (\hat{m}_\epsilon(i) - m_\epsilon(i))/\sqrt{\sigma^2_{\epsilon\epsilon}/N}$, where $m_\epsilon(i)$ is the population mean of ϵ, then the threshold for accepting the zero mean hypothesis is defined as τ_i where it follows from Eq. 6.4 that

$$\tau_i = 1.96\sqrt{\frac{\hat{\sigma}^2_{\epsilon\epsilon}(i)}{N}}, \tag{6.5}$$

where $\hat{\sigma}^2_{\epsilon\epsilon}(i)$, which is the relevant term on the diagonal of the innovation covariance $R_{\epsilon\epsilon}$, is the sample variance[1] on ϵ.

$$\hat{\sigma}^2_{\epsilon\epsilon}(i) = \frac{1}{N}\sum_{i=1}^{N} \epsilon_i^2(t). \tag{6.6}$$

[1]Recall that the sample variance given by $\hat{\sigma}^2_{\epsilon\epsilon}(i)$ is the variance on ϵ where $\hat{\sigma}^2_{\epsilon\epsilon}(i)/N$ is the sample variance on the *mean* of ϵ.

6.3.2 Innovation Sequence Whiteness Test

The whiteness test checks whether the innovation sequence is uncorrelated. The sample covariance function is given by

$$\hat{R}_{\epsilon\epsilon}(i,k) = \frac{1}{N}\sum_{t=k+1}^{N}[\epsilon_i(t) - \hat{m}_\epsilon(i)][\epsilon_i(t+k) - \hat{m}_\epsilon(i)]'. \tag{6.7}$$

This is now normalized such that

$$\hat{\rho}_{\epsilon\epsilon}(i,k) = \frac{\hat{R}_{\epsilon\epsilon}(i,k)}{\hat{R}_{\epsilon\epsilon}}. \tag{6.8}$$

For $N > 30$ this can be considered to be Gaussian with zero mean and variance $1/N$. Defining

$$W_{\epsilon\epsilon} = \hat{\rho}_{\epsilon\epsilon}(i,k) \pm \frac{1.96}{\sqrt{N}}, \tag{6.9}$$

and using Eq. 6.5, it follows that when 95 % of the values of $\hat{\rho}_{\epsilon\epsilon}(i,k)$ fall within this interval the whiteness criterion is met.

Although this whiteness test is useful for detecting model deficiencies for individual innovation sequence components, it becomes impractical when the number of measurements is large. This leads to a statistic containing all of the components called the weighted sum square residual or WSSR test. It accumulates the innovations vector over a finite window of length N_w and is defined as

$$\hat{\rho}(n) = \sum_{k=n-N_w+1}^{n} \epsilon'(k)R_{\epsilon\epsilon}^{-1}\epsilon(k). \quad n \geq N_w. \tag{6.10}$$

Derived in a manner similar to that for the zero mean test, the threshold τ for the 95 % interval test is defined by

$$\tau = N_y N_w + 1.96\sqrt{2N_y N_w}. \tag{6.11}$$

This provides a sliding window of width N_w when $n > N_w$.

As a final comment, the WSSR can be used as a detector in the sense that it will provide a detection statistic for the case when the model deviates from the information in the data. This is not surprising since Eq. 6.10 has the form of a log likelihood. For an example of this, see [2] in which a Kalman filter was tuned to the speckle noise from a laser Doppler vibrometer which was measuring the ground surface displacement as a means of detecting a buried mine. The speckle noise was modeled with an autoregressive model and the Kalman filter was tuned. When the

buried mine was vibrating at its resonant frequency due to an impulsive stimulus, the resonant vibration of the mine induced a small oscillatory displacement in the ground surface that was not contained in the autoregressive model of the speckle noise, hence the deviation from whiteness was detected easily by the WSSR test.

References

1. Candy, J.V.: Model-Based Signal Processing. Wiley, Hoboken (2006)
2. Sullivan, E.J., Xiang, N., Candy, J.V.: Adaptive model-based mine detection/localization using noisy laser doppler vibration. In: IEEE Oceans 09 Conference Proceedings, Bremen, Germany, 11–14 May, 2009

Index

E.J. Sullivan, *Model-Based Processing for Underwater Acoustic Arrays*,
SpringerBriefs in Physics, DOI 10.1007/978-3-319-17557-7