Kalman filtering and LQG-control in PRIMAL

van Meurs, P.

Award date:
1988

VAKGROEP SYSTEEM-
EN REGELTECHNIEK

NR-1513P (1988-02-11) P. van Meurs

KALMAN FILTERING AND LQG-CONTROL IN PRIMAL

P. van Meurs

AFSTUDEERVERSLAG

Afstudeerdocent: Prof.ir. O. Rademaker
Begeleider: Ir. R.J.P. van der Linden

ABSTRACT

Five modules have been developed to incorporate an LQG-control algorithm into PRIMAL (Package for Real time Interactive Modeling Analysis and Learning). This LQG-algorithm can be seperated into two independent sub-problems. Optimal state estimation and optimal state feedback.

Most of the effort has been put into developing 4 modules that can be used to design and test a steady-state Kalman filter. This Kalman filter is the best linear estimator for a system that can be described by a linear state space model with Gaussian white noise disturbances. The 4 modules are: KALSIM, a module that simulates a stochastic state space model, KALMAN, a module that calculates the Kalman filter, KALEST, a module that estimates the state vector for a stochastic state space model using a Kalman gain matrix and WTEST, a module that tests a signal on being zero mean white noise. A Kalman filter is functioning optimally when the residual vector (the difference between the predicted and real measurement) is zero mean white noise and the by KALMAN predicted variance is equal to the by SIMEST observed one. For cases of a sub-optimal Kalman filter, several approaches to adaptive filtering have been considered. The most promising one uses the autocorrelation function of the residuals to obtain better estimates for the noise components and the Kalman gain matrix. A simple SISO system is used to demonstrate the set of modules and the adaptive filtering approach.

The steady-state solution to the optimal state feedback matrix is obtained by solving the algebraic Riccati equation. This, numerically troublesome equation, is solved by obtaining the stable and unstable eigenvalues and eigenvectors of the corresponding Euler-Lagrange equation. The algorithm basically solves a generalized eigenvalue problem.

TABLE OF CONTENTS

LIST OF SYMBOLS

SYMBOL	FORMULA	SIZE	DESCRIPTION
δ_{ij}	3.3a	1	delta function
λ	6.8	1	multiplication factor
π	5.13a	*n-by-n*	covariance matrix
ρ_k	5.5	1	nornalized correlation function
σ_{max}	-	1	maximum eigenvalue
σ_{min}	-	1	minimum eigenvalue
ρ_k	5.5	1	normalized autocorrelation coefficient
Γ_k	5.21	*p-by-p*	innovation correlation matrix
$\mathbf{e}_1$	5.22a	*n*	estimation error
$\mathbf{e}_{est}$	3.13	*n*	estimation error
k	-	1	condition number
k	3.1	1	time index
n	3.1	1	dimension state vector
$\mathbf{n}_k$	2.1a	*t*	state noise process
m	3.1	1	dimension input vector
$\mathbf{m}_k$	2.1b	*p*	output noise process
p	3.2	1	dimension output vector
$\mathbf{q}_k$	5.44	*n*	estimate for $\mathbf{w}_k$
r	3.1	1	dimension state noise vector
$\mathbf{r}_k$	3.10	*p*	innovation/residual vector
$\mathbf{u}_k$	1.2	*m*	input vector
$\mathbf{v}_k$	2.2b	*p*	output noise vector
$\mathbf{w}_k$	2.2a	*r*	state noise vector
$\mathbf{x}_k$	1.2	*n*	state vector
$\mathbf{x}_0$	3.4a	*n*	initial state vector
$\hat{\mathbf{x}}_{est}$	3.13	*n*	estimated state vector
$\hat{\mathbf{x}}_{k+1/k}$	3.5	*n*	predicted state vector
$\hat{\mathbf{x}}_{k/k}$	3.8	*n*	estimated state vector
$\mathbf{y}_k$	2.1b	*p*	output vector
A	2.1a	*n-by-n*	system matrix
B	2.1a	*n-by-m*	input weight matrix
C	2.1b	*p-by-n*	output weight matrix
$\mathbf{C}_k$	-	-	autocorrelation function
$\hat{\mathbf{C}}_k$	5.3	-	estimate for $\mathbf{C}_k$
D	2.1b	*p-by-m*	input-output matrix
$E\{.\}$	1.1	1	expectation value
E	1.1	*n-by-n*	state weight matrix
F	1.1	*m-by-m*	input weight matrix
G	2.1a	*n-by-r*	noise weight matrix
H	5.9	*(pn)-by-n*	observability matrix
I	-	-	identity matrix
$\underset{\sim}{\mathbf{IR}}_k$	4.17	*p-by-p*	output covariance
$\hat{\mathbf{IR}}_k$	4.18	*p-by-p*	output covariance
J	1.2	1	cost function

$\mathbf{K}$	4.22	*n-by-p*	Kalman gain matrix
$\hat{\mathbf{K}}$	3.9	*n-by-p*	Kalman weight matrix
$\mathbf{L}$	1.2	*m-by-n*	state feedback matrix
$\mathbf{M}$	2.15	*n-by-n*	weight matrix
$\mathbf{M}$	6.8	*2n-by-2n*	matrix
$\mathbf{M}_{k+1}$	4.13	*n-by-p*	weight matrix
$\mathbf{N}$	6.9	*2n-by-2n*	mxtrix
$\mathbf{P}$	5.8	*n-by-n*	state covariance matrix
$\mathbf{P}_0$	3.4b	*n-by-n*	initial state covariance
$\mathbf{P}_{k+1/k}$	3.6	*n-by-n*	conditional covariance
$\mathbf{P}_{k/k}$	3.6	*n-by-n*	conditional covariance
$\mathbf{Q}$	3.3a	*r-by-r*	state noise covariance
$\mathbf{R}$	3.3b	*p-by-p*	output noise covariance
$\mathbf{S}_{k+1/k}$	4.5b	*n-by-n*	square root of $\mathbf{P}_{k+1/k}$
$\mathbf{S}_{k/k}$	4.5a	*n-by-n*	square root of $\mathbf{P}_{k/k}$
$\tilde{\mathbf{S}}_{k+1/k}$	4.8	*n-by-(n+r)*	state covariance
$\mathbf{T}$	4.9	*(n+r)-by-(n+r)*	transformation matrix
$\mathbf{T}'$	4.20	*(n+r)-by-(n+r)*	transformation matrix
$\mathbf{V}_k$	4.5d	*r-by-r*	square root of $\mathbf{Q}_k$
$\mathbf{W}$	5.13b	*p-by-p*	covariance of residuals
$\mathbf{W}_k$	4.5c	*p-by-p*	square root of $\mathbf{R}_k$
$\mathbf{X}$	5.23	*n-by-n*	covariance matrix

CHAPTER 1

INTRODUCTION

1.1. OPTIMAL CONTROL AND STATE ESTIMATION

Optimal control and state estimation are two closely related subjects but are two completely independent problems. This may seem to be a paradox, but when we look closer at what is meant by this remark it describes how a LQG (Linear Quadratic Gaussian) algorithm has been implemented.

1.1.1. THE LQG ALGORITHM

A LQG algorithm calculates an optimal feedback matrix **L**:

$$\mathbf{u}_k = \mathbf{L}\hat{\mathbf{x}}_{k/k-1} \qquad 1.1$$

$\mathbf{u}_k$ is the input vector at instant k, $\hat{\mathbf{x}}_k$ the estimate of the state variable $\mathbf{x}_k$ obtained by the Kalman filter.
L is optimal in the sense that it minimizes a quadratic cost function of the form:

$$J = E\{\sum_{k=0}^{\infty} [\mathbf{x}_k^T\mathbf{E}\mathbf{x}_k + 2\mathbf{u}_k^T\mathbf{M}\mathbf{x}_k + \mathbf{u}_k^T\mathbf{F}\mathbf{u}_k]\} \qquad 1.2$$

with **E** a positive semi-definite symmetric matrix, **M** positive semidefinite and **F** positive definite symmetric.

The system itself can be represented by a linear model corrupted by Gaussian white noise processes. The idea is to extract from the available input and output data an optimal estimate of the state $\hat{\mathbf{x}}$ of the system. Optimal in the sense that it performs better than any other algorithm and minimizes a certain cost function. Later it will be discussed what kind of cost functions are minimized.
The estimation problem (estimating the state variable) and the control problem (calculating the feedback matrix) are two completely independent problems in the sense that the feedback can be performed using the optimal estimate of state, which can be calculated separately from the feedback matrix. Together the two methods provide us with a powerful method for controlling a process that can be modeled by a linear model corrupted by independent Gaussian white noise processes. These assumptions do in no way restrain the practical use of the method as will be discussed in (ch.2.2). Since the two problems are completely independent we will first pay attention to estimation problems in general and to the Kalman filter as an optimal linear estimator in special. How to solve (1.2) resulting in a solution for **L** will be discussed in the chapter about optimal control and the Riccati equation.

1.1.2. STATE ESTIMATION, A MATHEMATICAL TOOL

All applications in PRIMAL calculate "black box" models from available input and output data. In this case state variables are just mathematical tools to model a process. Since there is a wide variety of mathematical tools and theory available for state estimation and state feedback we will use the concept of state variables even if they have no physical reality.

1.1.3. FINAL REMARKS

Since the strength of a control application is determined by the weakest component, it is important to put a lot of effort into developing a package that can design a Kalman filter and test and evaluate its performance and that can adapt the filter in case of sub-optimal behaviour. The second step, the implementation of the optimal feedback matrix has therefore been restricted to the algorithm for solving the Riccati equation. There are no results that show a control routine "in action" but there are results showing how to design an optimal Kalman filter and how to evaluate its performance.

CHAPTER 2

INTRODUCTION TO STOCHASTIC MODELS AND ESTIMATION

2.1. STOCHASTIC MODELS

Most of the techniques for designing optimal controllers are based upon deterministic models. However, these models are not always succesful in a noisy or uncertain environment. There are three reasons why we should consider stochastic models.

(1) Deterministic models are never perfect. The purpose of a model is to represent certain characteristics of the reality. Sometimes it is not possible to derive a complete model, since this would require a model with inifinite variables. Therefore, for practical and computational reasons, the model must be an approximation.

(2) There are stochastic disturbances. Noise processes and uncertainties in the model introduce a random component.

(3) Sensors do not provide perfect and complete data about the system. Sensors introduce their own dynamics and distortions in the measurements. Sometimes it is not possible to measure a variable directly.

Since we are interested in developing an optimal controller for a system, corrupted with noise and uncertainties, we are left with the following tasks:

(1) To develop a system model that accounts for these uncertainties and disturbances in a direct but practical way.

(2) To estimate the quantities of interest.

(3) To control the system in an optimal sense.

We will develop a stochastic model that accounts for uncertainties and disturbances in such a way that it can be used in practice.

2.1.1. THE STOCHASTIC SYSTEM MODEL

A potentially powerful and useful model is a discrete linear state equation:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{G}\mathbf{n}_k \qquad 2.1a$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{m}_k \qquad 2.1b$$

This model is an extension to the deterministic state space model, obtained by adding a noise process $\mathbf{n}_k$ to the dynamics equation and $\mathbf{m}_k$ to the output equation. If we would like to evaluate the probability function for $\mathbf{x}_k$ and for $\mathbf{y}_k$ in a practical way, we will

have to restrict the inputs to the ones obtained from Gauss-Markov processes. For a Markov process, the probability function at time-instant k completely determines the probability function at instants larger than k. In the context of linear models, it can be shown that the Markov assumption is equivalent to stating that the random process $\mathbf{n}_k$ and $\mathbf{m}_k$ are expressible as the outputs of linear models, called shaping filters, driven by white noises and deterministic inputs only.
Since a Gaussian probability function is completely determined by the mean and variance, and since linear operations on Gaussian processes result in Gaussian processes, we only have to propagate in time the first two moments to obtain a propagation of the complete probability function.
The model is thus reduced to a linear model of a physical system driven by deterministic inputs, white Gaussian noises and Gauss-Markov processes. Consequently, one can consider the original model and the necessary shaping filters as a single "augmented" linear system driven only by white Gaussian noises and deterministic inputs.

The model can thus be restricted to:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{G}\mathbf{w}_k \qquad 2.2a$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{v}_k \qquad 2.2b$$

where $\mathbf{x}_k$ is now the augmented system state and $\mathbf{w}_k$ and $\mathbf{v}_k$ are white Gaussian noises, assumed to be independent of each other (fig. 2.1).

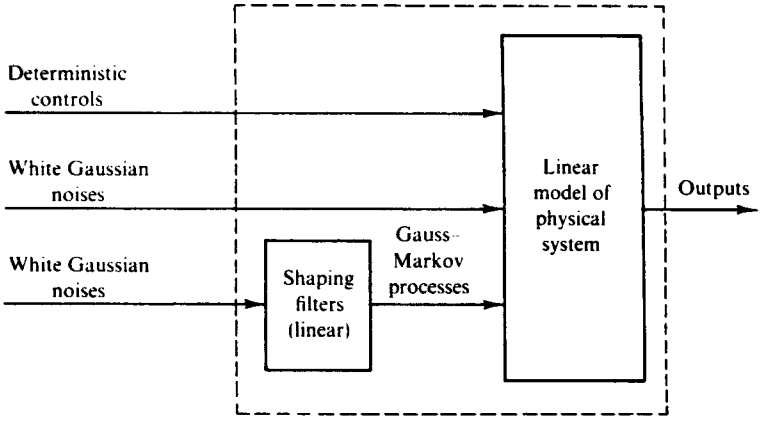

Fig. 2.1
The augmented system

We will now give an overview of estimation problems and the purpose of the KALman filter.

2.2. INTRODUCTION TO ESTIMATION PROBLEMS

A general estimation problem can be posed in the following manner. Given some quantities of interest whose values are not known exactly. Measuring devices and sensors provide data that is related to the variables of interest. The objective is to use this data, the knowledge about its relations to the quantities of interest and the knowledge about its noise-corruption to obtain an estimate of these variables. This estimate has to be "optimal" in a sense defined by some criterion.

This means that in general, an estimation problem has five fundamental components :

(1) the variables to be estimated
(2) the measurements or observations available
(3) the mathematical model describing how the measurements are related to the variables of interest
(4) the mathematical model of the uncertainties present
(5) the performance criterion to judge which estimation is "the best"

In our case we restrict ourselves to time (in)dependent, linear Gaussian systems. This means that the five components of our estimation problem can be specified as follows :

(1) The variables to be estimated are put in a n-dimensional state vector $\mathbf{x}_k$ (with k the time index) that are related as:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{G}\mathbf{w}_k$$

with $\mathbf{u}_k$, a m-dimensional, deterministic input vector.

(2) There will be a p-dimensional measurement vector $\mathbf{y}_k$.

(3) The set of measurements $\mathbf{y}_k$ are assumed to be a linear combination of the variables of interest, corrupted with a disturbance vector $\mathbf{v}_k$ of dimension p.:

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{v}_k \qquad 2.3$$

(4) The possible values of $\mathbf{x}_k$ are realizations of a random Gaussian variable with mean $\hat{\mathbf{x}}_k$ and covariance $\mathbf{P}_k$. Also the set $\mathbf{y}_k$ can be viewed as a realization of a random Gaussian variable.

(5) With respect to the performance criterion, the Bayesian viewpoint is adopted. We want to generate a complete description of the probability distribution for the values of the variables of interest. Since we are interested in estimating the values of a variable $\mathbf{x}_k$, knowing the measurements $\mathbf{y}_k$, we need to generate a conditional density function. Once we know this density function, it provides us with all the information necessary to define an optimal estimate, since reasonable definitions of optimality might include (fig. 2.2) the median (having equal weight on either side), the mode (maximum likelihood value), or the mean (the center of probability).

An algorithm that propagates the mean and covariance of the variables of interest and in case of Gaussian noise also the complete density function is called the Kalman filter.

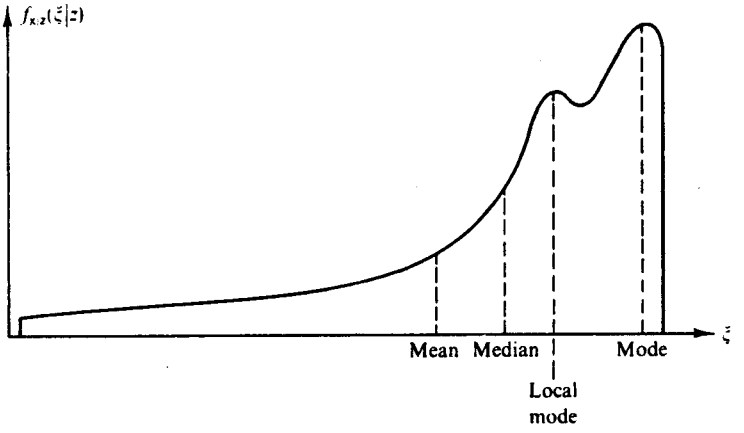

Fig. 2.2
Choice of estimator

2.2.1. THE KALMAN FILTER

Basically a Kalman filter is an *optimal recursive dataprocessing algorithm.* Although there are many ways of defining *optimal*, depending on the criterion chosen to evaluate performance, it can be shown that under some basic assumptions the Kalman filter is optimal with respect to virtual any sensible criterion.

The Kalman filter needs the following information to perform its function:

(1) A linear model of the system and measurement dynamics.
(2) A statistical description of the system noises, measurement noises and model uncertainties.
(3) Any available information about initial conditions of the variables of interest.

The word *recursive* means that the Kalman filter does not require all previous data to be kept in store nor does it need to reprocess all information every time a new measurement is taken. This makes the Kalman filter easy to implement.

A Kalman filter basically combines all available measurement data, plus knowledge about the system and measuring devices and noise statistics, to produce an estimate of a desired variable in such a way that the error is minimized statistically. This means that the average results of the Kalman filter are better than the average results of any other method. The next section will give a very elementary description of the necessary assumptions. In chapter 3 they will be treated more accurately.

2.2.2. BASIC ASSUMPTIONS

The Kalman filter will only give an optimal estimate if three basic assumptions are satisfied. They are:

(1) The model description is a linear model.

(2) The system and measurement noise are white.

(3) The system and measurement noise are Gaussian.

Some of these restrictions can be relaxed. For instance, if the Gaussian assumption is removed, the Kalman filter is still the best (minimum error variance) filter out of the class of linear filters.
These three assumptions do in no way restrain the practical use of the Kalman filter.

(1) A linear model is often adequate enough to describe the dynamics of a system. If the system contains non-linearities, the standard approach is to linearize about a chosen point or trajectory. Linear models are easier to manipulate mathematically and the linear theory is more complete and practical than the non-linear.

(2) A white noise model is used, since the mathematics involved are simple and usable in practical systems. In cases where the noise power level is not constant over all frequencies, or in which the noise is time correlated, a white noise through a linear system can duplicate the necessary characteristics of the noise. This system, called a "shaping filter" is added to the overall system to obtain a system driven by white noise again.

(3) The Gaussian property can be justified physically by the fact that system or measurement noise is caused by a number of small sources and it can be shown mathematically that when a number of independent random variables are added together, the summed effect can be adequately described by a Gaussian probability function, regardless of the shape of the individual densities. And since one knows, at best, the first and second order stochastics (mean and variance) of a noise process, the Gaussian assumption means that we can propagate in time the complete density function if we can propagate the mean and covariance.

CHAPTER 3

THE KALMAN FILTER

3.1. PROBLEM STATEMENT

We will describe the model that is used for the Kalman filter, the necessary information needed to solve the problem and the algorithm.

3.1.1. THE MODEL DESCRIPTION

Consider the following completely observable, completely controllable, discrete, linear, stochastic system:

$$\mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k + \mathbf{G}_k \mathbf{w}_k \quad 3.1$$

$$\mathbf{y}_k = \mathbf{C}_k \mathbf{x}_k + \mathbf{D}_k \mathbf{u}_k + \mathbf{v}_k \quad 3.2$$

where

$\mathbf{x}$ is a n-state vector
$\mathbf{u}$ is a m-input vector
$\mathbf{w}$ is a r-process noise vector
$\mathbf{y}$ is a p-measurement vector
$\mathbf{v}$ is a p-measurement vector

and

$\mathbf{A}$ is a n-by-n state transition matrix
$\mathbf{B}$ is a n-by-m input weight matrix
$\mathbf{G}$ is a n-by-r process noise matrix
$\mathbf{C}$ is a p-by-n output weight matrix
$\mathbf{D}$ is a p-by-m feedthrough matrix

3.1.2. THE STATISTICAL DESCRIPTION OF THE NOISE

The noise sequences $\mathbf{w}_k$ and $\mathbf{v}_k$ are uncorrelated Gaussian white noises with the following properties:

$$E\{\mathbf{w}_k \mathbf{w}_j^T\} = \mathbf{Q}_k \delta_{kj} \quad 3.3a$$

$$E\{\mathbf{v}_k \mathbf{v}_j^T\} = \mathbf{R}_k \delta_{kj} \quad 3.3b$$

$$E\{\mathbf{v}_k\} = \mathbf{0} \quad 3.3c$$

$$E\{\mathbf{w}_k\} = \mathbf{0} \quad 3.3d$$

$$E\{\mathbf{w}_k \mathbf{v}_k^T\} = \mathbf{0} \quad 3.3e$$

δ_{kj} is the delta function defined to be 1 if k=j and 0 if k≠j. $\mathbf{Q}_k$ is r-by-r symmetric, positive semidefinite matrix and $\mathbf{R}_k$ is p-by-p symmetric, positive definite matrix. Positive definiteness of $\mathbf{R}_k$ implies that all components of the measurement vector are corrupted by noise and there are no linear combinations of measurement vectors that are noise-free.

3.1.3. THE A PRIORI INFORMATION

(Eqn. 3.1) is propagated from the initial condition $\mathbf{x}_0$. However, since this value may not be known precisely a priori, it will be modeled as a random variable with a normal distribution. $\mathbf{x}_0$ is completely described by its mean $\hat{\mathbf{x}}_0$ and covariance $\mathbf{P}_0$:

$$E\{\mathbf{x}_0\} = \hat{\mathbf{x}}_0 \quad 3.4a$$

$$E\{[\mathbf{x}_0 - \hat{\mathbf{x}}_0][\mathbf{x}_0 - \hat{\mathbf{x}}_0]^T\} = \mathbf{P}_0 \quad 3.4b$$

where $\mathbf{P}_0$ is n-by-n symmetric, positive semidefinite matrix. A singular $\mathbf{P}_0$ means that some initial states or linear combinations of initial states are known precisely. Furthermore it is assumed that $\hat{\mathbf{x}}$, $\mathbf{w}_k$, $\mathbf{v}_k$ are uncorrelated.

3.1.4. THE CRITERION

The Kalman filter will be derived in a Bayesian manner by generating recursive equations to propagate in time the Gaussian conditional probability function and the mean value for the states [MAY79], [LEE64]. Another criterion is minimization of the mean-squared estimation error [KAL60] but this approach leads to the same results.

We now have a linear model with white noise disturbances and *a priori* information, enough information for the Kalman filter to combine the measurement data with the information provided by the system model and the statistical description of the uncertainties to obtain an optimal estimate of the system state.

3.2. THE ALGORITHM

We will only summarize the final algorithm, the interested reader can refer to the numerous publications about the Kalman filter [KAL60], [KAL61], [LEE64], [SOR85].

The optimal state estimate is propagated from instant k to k+1 by the relations [BRA75], [MAY79]:

TIME UPDATE

$$\hat{\mathbf{x}}_{k+1/k} = \mathbf{A}_k \hat{\mathbf{x}}_{k/k} + \mathbf{B}_k \mathbf{u}_k \quad 3.5$$

$$\mathbf{P}_{k+1/k} = \mathbf{A}_k \mathbf{P}_{k/k} \mathbf{A}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T \quad 3.6$$

Since no measurement is available, the best prediction for the mean state is what the model predicts (eqn. 3.5). The state estimation error $\mathbf{P}_{k/k}$ is propagated in time and the measurement uncertainty $\mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T$ is added, to obtain the prediction error $\mathbf{P}_{k+1/k}$ (eqn 3.6). At instant k+1 the measurement $\mathbf{y}_{k+1}$ becomes available. The estimate is *updated* by defining the Kalman gain matrix $\mathbf{K}_{k+1}$ and using it in the following relations

MEASUREMENT UPDATE

$$\hat{\mathbf{K}}_{k+1} = \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T [\mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T + \mathbf{R}_{k+1}]^{-1} \quad 3.7$$

$$\hat{\mathbf{x}}_{k+1/k+1} = \hat{\mathbf{x}}_{k+1/k} + \hat{\mathbf{K}}_{k+1} [\mathbf{y}_{k+1} - \mathbf{C}_{k+1} \hat{\mathbf{x}}_{k+1/k} - \mathbf{D}_{k+1} \mathbf{u}_{k+1}] \quad 3.8$$

$$\mathbf{P}_{k+1/k+1} = \mathbf{P}_{k+1/k} - \hat{\mathbf{K}}_{k+1} \mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \quad 3.9$$

Where $\hat{\mathbf{x}}_{k+1/k}$ is the conditional mean of $\mathbf{x}_{k+1}$ before the measurement $\mathbf{y}_{k+1}$ is taken and processed and $\hat{\mathbf{x}}_{k+1/k+1}$ is the conditional mean of $\mathbf{x}_{k+1}$ after the measurement $\mathbf{y}_{k+1}$ has been processed. $\mathbf{P}_{k+1/k}$ and $\mathbf{P}_{k+1/k+1}$ are the conditional error covariances of $\mathbf{x}_{k+1}$ before respectively after the measurement is available.

The structure of the filter is as follows. The input to the algorithm is $\mathbf{y}_{k+1}$, the measurement. The measurement residual $\mathbf{r}_{k+1}$ is generated as the difference between the measurement $\mathbf{y}_{k+1}$ and the best prediction of it before it was taken:

$$\mathbf{r}_{k+1} = \mathbf{y}_{k+1} - \mathbf{C}_{k+1} \hat{\mathbf{x}}_{k+1/k} - \mathbf{D}_{k+1} \mathbf{u}_{k+1} \quad 3.10$$

This residual or innovation is important in adaptive algorithms and sensor failure detection. The residual is multiplied with an optimal weighting matrix $\hat{\mathbf{K}}_{k+1}$ to generate a correction term to be added to the predicted value of state $\hat{\mathbf{x}}_{k+1/k}$ resulting in a corrected value $\hat{\mathbf{x}}_{k+1/k+1}$. The algorithm has therefore a predictor-corrector structure.

A positive definite $\mathbf{R}$ and a positive semidefinite $\mathbf{Q}$ together with:

(1) $(\mathbf{A},\mathbf{C})$ completely observable (necessary condition).

(2) $(\mathbf{A},\mathbf{W})$ completely controllable, with $\mathbf{W}$ any n-by-n matrix such that $\mathbf{W}\mathbf{W}^T = \mathbf{Q}$ (sufficient condition).

CHAPTER 4

NUMERICAL PROBLEMS AND SQUARE ROOT MATRICES

4.1. INTRODUCTION

Despite the fact that the equations for solving the Kalman filter are well defined, there are some numerical difficulties in solving the equations. The most troublesome numerical aspect of the Kalman filter is the measurement update of the covariance matrix.

The update formula (eqn. 3.9) :

$$\mathbf{P}_{k+1/k+1} = \mathbf{P}_{k+1/k} - \hat{\mathbf{K}}_{k+1}\mathbf{C}_{k+1}\mathbf{P}_{k+1/k} \qquad 4.1$$

can, using (eqn. 3.7), be written as:

$$\mathbf{P}_{k+1/k+1} = \mathbf{P}_{k+1/k} - \mathbf{P}_{k+1/k}\mathbf{C}_{k+1}^T[\mathbf{C}_{k+1}\mathbf{P}_{k+1/k}\mathbf{C}_{k+1}^T + \mathbf{R}_{k+1}]^{-1}\mathbf{C}_{k+1}\mathbf{P}_{k+1/k} \qquad 4.2$$

Both equations have their own numerical difficulties. (Eqn. 4.2) involves p-by-p inversions, where p is the dimension of the measurement vector, but (eqn. 4.1) can involve small differences of large numbers, resulting in numerical problems on finite wordlength computers. This is especially the case when measurements are very accurate. On finite wordlength computers these numerical problems can result in a non-positive definite result, which is a theoretical impossibility.

The update can also be written as:

$$\mathbf{P}_{k+1/k+1} = [\mathbf{I} - \hat{\mathbf{K}}_{k+1}\mathbf{C}_{k+1}]\mathbf{P}_{k+1/k} \qquad 4.3$$

This form is, unlike (eqn. 4.2), not able to ensure positive definiteness and can additionally not preserve symmetry very well. (eqn. 4.1) and (eqn. 4.2) involve subtraction of two symmetric matrices whereas (eqn. 4.3) is in the form of a product of non-symmetric matrix and a symmetric one, which does not necessarily result in a symmetric matrix.
The consequences of these problems in calculating covariance measurement updates make it necessary to calculate in double precision to maintain numerical accuracy. For on-line applications, symmetry can be used by propagating and updating only lower or upper triangular forms of the covariance matrix. This requires only $\frac{1}{2}n(n+1)$ terms instead of n^2.
Symmetry can be exploited further by using a square root covariance formulation.

This means that the algorithm does not propagate **P** but $\mathbf{P}^{1/2}$. In case of a symmetric, positive definite matrix **P**, it is always possible to find a matrix $\mathbf{P}^{1/2}$ such that $\mathbf{P}^{1/2}\mathbf{P}^{1/2T} = \mathbf{P}$. Square root formulation either increases the numerical accuracy with a factor of two or makes it possible to achieve the same accuracy on a computer with half the wordlength. The prove of this is simple. The condition number, a concept often used to analyze the effect of perturbations in linear equations, of a matrix **A** is defined to be:

$$k(\mathbf{A}) = \sigma_{max}/\sigma_{min}$$

where σ^2_{max} and σ^2_{min} are the maximum and minimum eigenvalues of $\mathbf{A}^T\mathbf{A}$. When computing in base 10 arithmetic with N significant digits, numerical difficulties may be expected as $k(\mathbf{A})$ approaches 10^N [MAY79]. But,

$$k(\mathbf{P}) = k(\mathbf{S}\mathbf{S}^T) = [k(\mathbf{S})]^2$$

Therefore, while calculations on **P** may encounter difficulties when $k(\mathbf{P}) = 10^N$, those same numerical problems would arise when $k(\mathbf{S}) = 10^{N/2}$. This means that the same numerical precision is achieved with half the wordlength.
Square root algorithms also maintain a symmetric, positive semidefinite form of the covariance matrix. In the next section, the concept of matrix square roots is introduced.

4.2. MATRIX SQUARE ROOTS

Let **A** be an n-by-n, symmetric, positive definite matrix. Without proof it is assumed that there exists at least one n-by-n "square root" matrix $\sqrt{\mathbf{A}}$, such that

$$\mathbf{A}^{1/2}\mathbf{A}^{1/2T} = \mathbf{A} \qquad 4.4$$

In fact, there are many "square root" matrices which satisfy (eqn. 4.4) and this property will be used to take an attractive form.

By propagating the square root of the state error covariance matrix **P** instead of **P** itself, it is impossible for **P** to become negative. In a scalar case this means that instead of the variance σ^2, the standard deviation σ is propagated, this way reducing the numerical range by a factor of two and increasing the numerical accuracy by a factor of two.

To propagate the error covariance matrix **P** it is necessary to define the following square root matrices

$$\mathbf{S}_{k+1/k+1}\mathbf{S}^T_{k+1/k+1} = \mathbf{P}_{k+1/k+1} \qquad 4.5a$$

$$\mathbf{S}_{k+1/k}\mathbf{S}^T_{k+1/k} = \mathbf{P}_{k+1/k} \qquad 4.5b$$

$$\mathbf{W}_k\mathbf{W}^T_k = \mathbf{Q}_k \qquad 4.5c$$

$$\mathbf{V}_k\mathbf{V}^T_k = \mathbf{R}_k \qquad 4.5d$$

As stated before, these square roots are not uniquely defined by (eqn. 4.5a-d), but since square root filters can be formulated in terms of general matrix square roots it is not necessary to define the exact form of the matrices. By developing algorithms that use and maintain special forms of square roots, it is possible to minimize storage requirements and computation time. A specially attractive form is an upper or lower triangular matrix (e.g. a matrix with all zeros above or below the diagonal). This way only $\frac{1}{2}n(n+1)$ instead of n^2 variables need to be computed and stored. An algorithm that computes a unique lower triangular, square root matrix given a symmetric, positive definite matrix is called the Cholesky decomposition . To take care of positive semidefinite matrices, a row and column exchange can move the zero diagonal element to the end of the matrix and a size reduction of the matrix will ensure positive definiteness of this submatrix [MAY79].

4.3. THE ALGORITHM

We will use matrix square roots to calculate the numerically difficult time update for the state error covariance matrix [AND71].
The time update equation for the error covariance matrix is:

$$\mathbf{P}_{k+1/k} = \mathbf{A}_k \mathbf{P}_{k/k} \mathbf{A}_k^T + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^T \qquad 4.6$$

Using (eqn. 4.5b-c) this equation can also be written as

$$\mathbf{P}_{k+1/k} = \mathbf{A}_k \mathbf{S}_{k/k} \mathbf{S}_{k/k}^T \mathbf{A}_k^T + \mathbf{G}_k \mathbf{W}_k \mathbf{W}_k^T \mathbf{G}_k^T \qquad 4.7$$

Our objective is to find a propagation equation for the square root of $\mathbf{P}_{k+1/k}$. Therefore it is necessary to find a matrix $\mathbf{S}_{k+1/k}$ such that $\mathbf{S}_{k+1/k} \mathbf{S}_{k+1/k}^T$ is equal to the right hand side of (eqn. 4.7).

One such matrix $\tilde{\mathbf{S}}_{k+1/k}$ is obvious:

$$\tilde{\mathbf{S}}_{k+1/k} = [\mathbf{A}_k \mathbf{S}_{k/k} \mid \mathbf{G}_k \mathbf{W}_k] \qquad 4.8$$

Since $\mathbf{S}_{k/k}$ is n-by-n, then $\tilde{\mathbf{S}}_{k+1/k}$ is n-by-$(n+r)$. This means that every timestep the dimension of the square root matrix increases which must be rejected for being computationally impractical.
But if we can find an orthonormal $(n+r)$-by-$(n+r)$ transformation matrix $\mathbf{T}$ such that, $\tilde{\mathbf{S}}_{k+1/k} \mathbf{T}$ is a square root of $\mathbf{P}_{k+1/k}$, then:

$$\tilde{\mathbf{S}}_{k+1/k} \mathbf{T}\mathbf{T}^T \tilde{\mathbf{S}}_{k+1/k}^T = \tilde{\mathbf{S}}_{k+1/k} \tilde{\mathbf{S}}_{k+1/k}^T \qquad 4.9$$

This means that if it is possible to find an orthonormal matrix, such that

$$\tilde{\mathbf{S}}_{k+1/k} \mathbf{T} = [\mathbf{S}_{k+1/k} \mid 0] \qquad 4.10$$

we have found an n-by-n square root matrix $\mathbf{S}_{k+1/k}$ that satisfies the required relationships. If in addition $\mathbf{S}_{k+1/k}$ would be lower triangular, this would mean that the

algorithm propagates a lower triangular, square root matrix.
There are two methods that generate such $\mathbf{S}_{k+1/k}$ (known as triangularization algorithms) : Gram-Schmidt orthonormalization and Householder transformation.

We can write the final result as

$$[\mathbf{S}_{k+1/k} \mid 0] = [\mathbf{A}_k \mathbf{S}_{k/k} \mid \mathbf{G}_k \mathbf{W}_k] \mathbf{T} \tag{4.11}$$

The same procedure can also be applied to the measurement update equation for the error covariance matrix

$$\mathbf{P}_{k+1/k+1} = \mathbf{P}_{k+1/k} - \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T (\mathbf{R}_{k+1} + \mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T)^{-1} \mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \tag{4.12}$$

Since we also want to calculate the measurement update for the state vector a procedure to compute them simultaneously has been developed. We write (eqn. 3.8) as:

$$\hat{\mathbf{x}}_{k+1/k+1} = \hat{\mathbf{x}}_{k+1/k} + \mathbf{M}_{k+1} (\mathbf{R}_{k+1} + \mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T)^{-\frac{1}{2}} \mathbf{r}_{k+1} \tag{4.13}$$

with $\mathbf{M}_{k+1}$ defined as

$$\mathbf{M}_{k+1} = \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T (\mathbf{R}_{k+1} + \mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T)^{-\frac{1}{2}} \tag{4.14}$$

using (eqn. 4.14) equation (eqn. 4.12) is written as

$$\mathbf{S}_{k+1/k+1} \mathbf{S}_{k+1/k+1}^T + \mathbf{M}_{k+1} \mathbf{M}_{k+1}^T = \mathbf{P}_{k+1/k} \tag{4.15}$$

now we want to find a square root relation for the right hand side which is equal to $\mathbf{S}_{k+1/k}$ (eqn. 4.15).
One such matrix would be

$$[\mathbf{S}_{k+1/k+1} \mid \mathbf{M}_{k+1}] \tag{4.16}$$

A transformation matrix can be used to give $\mathbf{S}_{k+1/k+1}$ a special form, like upper or lower triangular. To update the state vector we need to calculate the Kalman gain matrix $\mathbf{K}_{k+1}$, which means we need to know $\mathbf{M}_{k+1}$ and the inverse of the square root of $(\mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T + \mathbf{R}_{k+1})$. The latter can be written as

$$\mathbf{IR}_{k+1} \mathbf{IR}_{k+1}^T = \mathbf{V}_{k+1} \mathbf{V}_{k+1}^T + \mathbf{C}_{k+1} \mathbf{S}_{k+1/k} \mathbf{S}_{k+1/k}^T \mathbf{C}_{k+1}^T \tag{4.17}$$

with $\mathbf{IR}_{k+1}$ the square root of $\mathbf{R}_{k+1} + \mathbf{C}_{k+1} \mathbf{P}_{k+1/k} \mathbf{C}_{k+1}^T$.

One such matrix $\tilde{\mathbf{IR}}_{k+1}$ can be written as

$$\tilde{\mathbf{IR}}_{k+1} = [\mathbf{V}_{k+1} \mid \mathbf{C}_{k+1} \mathbf{S}_{k+1/k}] \tag{4.18}$$

Since we can choose an orthonorm transformation matrix $\mathbf{T}$ freely, we choose one that

preserves the dimensions. So if we can find an orthonormal matrix **T** such that

$$\tilde{\mathbf{IR}}_{k+1}\mathbf{T} = [\mathbf{IR}_{k+1} \mid 0] \tag{4.19}$$

then we have found an p-by-p square root matrix $\mathbf{IR}_{k+1}$ which satisfies the desired relationship.

Now we can combine the measurement update for the covariance matrix and for the state vector

$$\begin{bmatrix} (\mathbf{R}_{k+1} + \mathbf{C}_{k+1}\mathbf{P}_{k+1/k}\mathbf{C}_{k+1}^T)^{1/2} & 0 \\ \mathbf{M}_{k+1} & \mathbf{S}_{k+1/k+1} \end{bmatrix} = \begin{bmatrix} \mathbf{V}_{k+1} & \mathbf{C}_k\mathbf{S}_{k+1/k} \\ 0 & \mathbf{S}_{k+1/k} \end{bmatrix} \mathbf{T}^* \tag{4.20}$$

Now we can combine the time update (eqn. 4.11) and the measurement update (eqn. 4.20) as follows

$$\begin{bmatrix} \mathbf{V}_k & \mathbf{C}_k\mathbf{S}_{k/k-1} & 0 \\ 0 & \mathbf{A}_k\mathbf{S}_{k/k-1} & \mathbf{G}_k\mathbf{W}_k \end{bmatrix} \begin{bmatrix} \mathbf{T}^* & 0 \\ 0 & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & 0 \\ 0 & \mathbf{T} \end{bmatrix}$$

$$= \begin{bmatrix} (\mathbf{R}_k + \mathbf{C}_k\mathbf{P}_{k/k-1}\mathbf{C}_k^T)^{1/2} & 0 & 0 \\ \mathbf{A}_k\mathbf{M}_k & \mathbf{A}_k\mathbf{S}_{k/k} & \mathbf{G}_k\mathbf{W}_k \end{bmatrix} \begin{bmatrix} \mathbf{I} & 0 \\ 0 & \mathbf{T} \end{bmatrix} \tag{4.21}$$

$$= \begin{bmatrix} (\mathbf{R}_k + \mathbf{C}_k\mathbf{P}_{k/k-1}\mathbf{C}_k^T)^{1/2} & 0 & 0 \\ \mathbf{A}_k\mathbf{M}_k & \mathbf{S}_{k+1/k} & 0 \end{bmatrix}$$

The **T** and $\mathbf{T}^*$ are Householder transformation matrices which maintain the special form of the covariance matrices without an increase in dimensions. The Kalman gain matrix $\hat{\mathbf{K}}_k$ is given by:

$$\mathbf{K}_k = \mathbf{A}_k\hat{\mathbf{K}}_k = \mathbf{A}_k\mathbf{M}_k(\mathbf{R}_k + \mathbf{C}_k\mathbf{P}_{k/k-1}\mathbf{C}_k)^{-1/2} \tag{4.22}$$

Note the difference between $\hat{\mathbf{K}}$, the Kalman gain matrix, and **K**, the Kalman gain matrix multiplied by the system matrix **A**. In the literature the definition of the Kalman gain matrix can be either of these. We will use **K** as the Kalman gain matrix since the algorithm provides us with it.

CHAPTER 5

ADAPTIVE FILTERING TECHNIQUES

5.1. UNKNOWN NOISE COVARIANCE MATRICES

It is not always possible to know beforehand the exact noise covariance matrices **Q** and **R**. Therefore we can not be sure that the calculated Kalman gain matrix is optimal. In certain cases, the use of wrong *a priori* statistics can lead to large estimation errors or even divergence. The purpose of adaptive filtering is to reduce the errors by adapting the Kalman filter to real data.
In PRIMAL only so called black-box models are developed. This means that there is little or no correspondence of the models with physical properties of the system. Some of the applications estimate the error component but not in the by Kalman required form since transformation of these matrix fraction description models into state space form results in a stochastic state space model with correlated $\mathbf{w}_k$ and $\mathbf{v}_k$.
Since the Kalman filter is only optimal if the noise covariance matrices are known exactly, it is important to develop methods to obtain these as accurate as possible. Several methods have been developed among which, output correlation methods, innovation correlation methods and covariance matching techniques.

5.2. SOME DEFINITIONS

The time independent, discrete time system is defined by the equations:

$$\mathbf{x}_{i+1} = \mathbf{A}\mathbf{x}_i + \mathbf{B}\mathbf{u}_i + \mathbf{G}\mathbf{w}_i \qquad 5.1$$

$$\mathbf{y}_i = \mathbf{C}\mathbf{x}_i + \mathbf{D}\mathbf{u}_i + \mathbf{v}_i \qquad 5.2$$

The *a priori* information and the noise covariance matrices are defined by (eqn. 3.3-4). Note that the time index is i instead of k, k will be used as the lag of the autocorrelation.
We use (eqn. 3.8 and eqn. 4.22) to obtain:

$$\hat{\mathbf{x}}_{i+1/i} = \mathbf{A}\hat{\mathbf{x}}_{i/i-1} + \mathbf{B}\mathbf{u}_i + \mathbf{K}(\mathbf{y}_i - \mathbf{C}\mathbf{x}_{i/i-1} - \mathbf{D}\mathbf{u}_i)$$

5.3. OPTIMALITY TESTING

An interesting problem of adaptive filtering is deciding whether the Kalman gain matrix **K** constructed using some estimates of **Q** and **R** is close to optimal or not (*hypothesis testing*).
It is sufficient for the optimality of a Kalman gain matrix that the innovation sequence $\mathbf{r}_i$ is white and of zero mean. The last condition is often forgotten but is of essential importance [MAR74]. There are many statistical tests for testing the whiteness of a sequence. One suited for larger datasets will be discussed. We will use the autocorrelation function $\mathbf{C}_k$ to test the whiteness for a sequence.
In this method we obtain an estimate of $\mathbf{C}_k$, denoted as $\hat{\mathbf{C}}_k$, by using the ergodic property of a stationary random sequence. A process is ergodic if any statistic calculated by averaging over all members of the ensemble of samples at a fixed time can be calculated equivalently by time-averaging over any single representative member of the ensemble, except a single member out of a set of probability zero.

$$\hat{\mathbf{C}}_k = \frac{1}{N}\sum_{i=k}^{N} \mathbf{r}_i \mathbf{r}_{i-k}^T \qquad 5.3$$

where N is the number of sample points and k the lag.
The estimates $\hat{\mathbf{C}}_k$ are biased for finite sample sizes:

$$E\{\hat{\mathbf{C}}_k\} = \left(1 - \frac{k}{N}\right)\mathbf{C}_k \qquad 5.4$$

However, it can be shown that the estimate of (eqn. 5.3) is prefered to the corresponding unbiased estimate, since it has a lower variance.
Estimates of the normalized autocorrelation coefficients ρ_k are obtained by dividing the elements of $\hat{\mathbf{C}}_k$ by the appropriate elements of $\hat{\mathbf{C}}_0$.

$$[\rho_k]_{ij} = \frac{[\hat{\mathbf{C}}_k]_{ij}}{[\hat{\mathbf{C}}_0]_{ii}[\hat{\mathbf{C}}_0]_{jj}^{1/2}} \qquad 5.5$$

It can be proven that $[\rho_k]_{ii}$ has an asymptotically normal distribution. Therefore, the 95 percent confidence limits for $[\rho_k]_{ii}$ are $\pm (1.96/N^{1/2})$.
To test the whiteness of a sequence, look at a set of values for $[\rho_k]_{ii}$ $(k > 0)$ and check the number of times they lie outside the band $\pm (1.96/N^{1/2})$. If this number is less than 5 percent of the total, the sequence $\mathbf{r}_i$ will be called white. This test is based on the assumption of large N. If N is small, other tests can be used [AND51], [HAN51].

5.4. CORRELATION METHODS

Correlation methods are used extensively in time series analysis. The basic idea is to correlate the output of a system either directly or after a known linear operation on it. A set of equations is derived relating system parameters to the calculated autocorrelation functions and these equations are solved for the unknown parameters. Two different methods can be developed considering either the autocorrelation

function of the output **y** or the autocorrelation function of the innovations **r**. The estimates obtained from the latter method turn out to be more efficient than the first since innovations are less correlated than outputs. Furthermore, the first method is only usable if the output **y** is stationary or **A**, the system matrix, is a stable matrix. The second method can also be used if these conditions are not satisfied. As it is necessary for the whiteness test to calculate the autocorrelation functions of the innovation sequence, the second method is preferable. Both methods assume complete controllability and observability of the system.

5.4.1. OUTPUT CORRELATION METHOD

Let $\mathbf{C}_k$ be the kth lag autocorrelation of the output $\mathbf{y}_i$:

$$\mathbf{C}_k = E\{\mathbf{y}_i \mathbf{y}_{i-k}^T\} \tag{5.6}$$

It is assumed that the output is stationary so that the autocorrelation is only a function of the lag. An expression for $\mathbf{C}_k$ can easily be derived from (eqn. 5.1-5.2) and $\mathbf{D} = 0$:

$$\mathbf{C}_k = \begin{cases} \mathbf{C}\mathbf{P}\mathbf{C}^T + \mathbf{R} & k = 0 \\ \mathbf{C}\mathbf{A}^k\mathbf{P}\mathbf{C}^T & k > 0 \end{cases} \tag{5.7}$$

where $\mathbf{P} = E\{\mathbf{x}_i \mathbf{x}_i^T\}$ satisfies:

$$\mathbf{P} = \mathbf{A}\mathbf{P}\mathbf{A}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T \tag{5.8}$$

Combining (eqn. 5.7-8) we obtain an expression that can be solved for **Q** and **R** provided the system is observable. (Eqn. 5.7) can be written for $k = 1,\ldots,n$,

$$\begin{bmatrix} \mathbf{C}_1 \\ \vdots \\ \mathbf{C}_n \end{bmatrix} = \mathbf{H}\mathbf{P}\mathbf{C}^T \tag{5.9}$$

where $\mathbf{H}^T = \left[\mathbf{A}^T\mathbf{C}^T, \cdots, (\mathbf{A}^T)^n\mathbf{C}^T\right]$. Since **A** is non-singular and the system is observable, which means that $rank(\mathbf{H}) = n$, the set of equations can be solved.

Solving (eqn. 5.9) for $\mathbf{P}\mathbf{C}^T$,

$$\mathbf{P}\mathbf{C}^T = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T \begin{bmatrix} \mathbf{C}_1 \\ \vdots \\ \mathbf{C}_n \end{bmatrix} \tag{5.10}$$

using (eqn. 5.7) for $k = 0$.

$$\mathbf{R} = \mathbf{C}_0 - \mathbf{C}(\mathbf{P}\mathbf{C}^T) \tag{5.11}$$

To obtain **Q** we must solve (eqn. 5.8) and (eqn. 5.10). A unique solution for **Q** cannot be obtained, in general. Only if $r \leqslant n$ i.e. if the dimension of the process noise vector is equal or smaller than the dimension of the state vector, a unique solution for **Q** exists. However, the optimal steady-state Kalman filter gain **K** can still be determined uniquely as follows.

Define

$$\pi = E\{\hat{\mathbf{x}}_{i/i-1}\hat{\mathbf{x}}_{i/i-1}^T\}$$

from (eqn. 3.5-3.8)

$$\hat{\mathbf{x}}_{i+1/i} = \mathbf{A}\hat{\mathbf{x}}_{i/i-1} + \mathbf{B}\mathbf{u}_i + \mathbf{K}\mathbf{r}_i \tag{5.12}$$

with $\mathbf{r}_i$, the innovation sequence, defined by (eqn. 3.10) and $\mathbf{u}_k$, the deterministic input vector.

Therefore,

$$\pi = \mathbf{A}\pi\mathbf{A}^T + \mathbf{K}\mathbf{W}\mathbf{K}^T \tag{5.13a}$$

with

$$\mathbf{W} = (\mathbf{C}\mathbf{P}\mathbf{C}^T + \mathbf{R}) \tag{5.13b}$$

W is the covariance matrix of the innovation or residual sequence $\mathbf{r}_i$.

Use has been made of the fact that $\mathbf{r}_i$ is white and uncorrelated to $\hat{\mathbf{x}}_{i/i-1}$. Another relationship for π can be derived from the fact that

$$\mathbf{x}_i = \hat{\mathbf{x}}_{i/i-1} + \mathbf{e}_i \tag{5.14}$$

where the error $\mathbf{e}_i$ has covariance **M** and is uncorrelated to $\hat{\mathbf{x}}_{i/i-1}$ because of the orthogonality principle (the error vector contains no more information for the estimated state vector):

$$E\{\hat{\mathbf{x}}_{i/i-1}\mathbf{e}_i^T\} = 0 \tag{5.15}$$

Therefore,

$$\mathbf{P} = \pi + \mathbf{M} \tag{5.16}$$

Now the Kalman gain **K** is given by (eqn. 3.7):

$$\begin{aligned} \mathbf{K} &= \mathbf{A}\mathbf{M}\mathbf{C}^T\mathbf{W}^{-1} \\ &= \mathbf{A}(\mathbf{P} - \pi)\mathbf{C}^T(\mathbf{C}_0 - \mathbf{C}\pi\mathbf{C}^T)^{-1} \end{aligned} \tag{5.17}$$

substituting for **K** in (eqn. 5.13a-b),

$$\pi = \mathbf{A}[\pi + (\mathbf{P} - \pi)\mathbf{C}^T(\mathbf{C}_0 - \mathbf{C}\pi\mathbf{C}^T)^{-1}\mathbf{C}(\mathbf{P} - \pi)]\mathbf{A}^T \qquad 5.18$$

which can be solved since $\mathbf{PC}^T$ is known from (eqn. 5.10).
Then **K** is obtained from (eqn. 5.17).

We have now derived all the equations necessary to calculate **Q**, **R** and **K** from the autocorrelations $\mathbf{C}_0, \cdots, \mathbf{C}_n$. The autocorrelations are estimated as:

$$\hat{\mathbf{C}}_k^N = \sum_{i=k}^{N} \mathbf{y}_i \mathbf{y}_{i-k}^T \qquad 5.19$$

where N is defined to be the sample size.

It can be shown that the estimates $\hat{\mathbf{C}}_k$ are asymptotically normal and unbiased. Since **Q** and **R** are linearly related to $\hat{\mathbf{C}}_k$, their estimates are also assymptotically normal and unbiased. Similarly, the estimate of **K** is assymptotically unbiased. This method is extensively discussed in [MEH70]. The estimates for the Kalman gain matrix and the noise matrices can be improved until the Kalman filter behaves optimal, as tested with the in section 5.3 developed method.

5.4.2. INNOVATION CORRELATION METHOD

From the theory of Kalman filtering it is known that the innovation sequence :

$$\mathbf{r}_i = \mathbf{y}_i - \mathbf{C}\hat{\mathbf{x}}_{i/i-1} - \mathbf{D}\mathbf{u}_i = \mathbf{C}\mathbf{e}_i + \mathbf{v}_i \qquad 5.20a$$

with $\mathbf{e}_i$ defined as:

$$\mathbf{e}_i = \mathbf{x}_i - \hat{\mathbf{x}}_{i/i-1} \qquad 5.20b$$

is a zero mean, Gaussian white noise sequence for an optimal filter. However, for a suboptimal filter, the innovation sequence is correlated as will be shown below.

Let

$$\Gamma_k = E\{\mathbf{r}_i \mathbf{r}_{i-k}^T\}$$

substituting from (eqn. 5.20a) and considering $k > 0$,

$$\begin{aligned}\Gamma_k &= E\{(\mathbf{C}\mathbf{e}_i + \mathbf{v}_i)(\mathbf{C}\mathbf{e}_{i-k} + \mathbf{v}_{i-k})^T\} \\ &= \mathbf{C}E\{\mathbf{e}_i \mathbf{e}_{i-k}^T\}\mathbf{C}^T + \mathbf{C}E\{\mathbf{e}_i \mathbf{v}_{i-k}^T\}\end{aligned} \qquad 5.21$$

since for $(k > 0)$, $\mathbf{v}_i$ is independent of $\mathbf{v}_{i-k}$ and $\mathbf{e}_{i-k}$.

and for $k = 0$,

$$\Gamma_0 = \mathbf{CMC}^T + \mathbf{R}$$

The expectation terms in (eqn. 5.21) can be evaluated by writing the difference equation for $\mathbf{e}_i$. We assume that the filter uses a suboptimal *a priori* gain $\hat{\mathbf{K}}_0$:

$$\mathbf{e}_i = \mathbf{A}(\mathbf{I} - \hat{\mathbf{K}}_0\mathbf{C})\mathbf{e}_{i-1} + \mathbf{A}\hat{\mathbf{K}}_0\mathbf{v}_{i-1} + \mathbf{G}\mathbf{w}_{i-1} \qquad 5.22$$

repeating this k steps backwards,

$$\mathbf{e}_i = [\mathbf{A}(\mathbf{I} - \hat{\mathbf{K}}_0\mathbf{C})]^k \mathbf{e}_{i-k} - \sum_{j=1}^{k} [\mathbf{A}(\mathbf{I} - \hat{\mathbf{K}}_0\mathbf{C})]^{j-1}\mathbf{A}\hat{\mathbf{K}}_0\mathbf{v}_{i-j} + \sum_{j=1}^{k} [\mathbf{A}(\mathbf{I} - \hat{\mathbf{K}}_0\mathbf{C})]^{j-1}\mathbf{G}\mathbf{w}_{i-j} \qquad 5.23$$

Therefore,

$$E\{\mathbf{e}_i \mathbf{e}_{i-k}^T\} = [\mathbf{A}(\mathbf{I} - \hat{\mathbf{K}}_0\mathbf{C})]^k \mathbf{M}_1 \qquad 5.24a$$

where

$$E\{\mathbf{e}_i \mathbf{e}_i^T\} = \mathbf{M}_1$$
$$E\{\mathbf{v}_{i-j} \mathbf{e}_{i-k}\} = 0, \quad j = 1,\ldots,k$$
$$E\{\mathbf{w}_{i-j} \mathbf{e}_{i-k}\} = 0, \quad j = 1,\ldots,k$$

because of the fact that the system is causal. $\mathbf{M}_1$ is the state error covariance matrix belonging to the suboptimal filter with Kalman gain $\hat{\mathbf{K}}_0$.

similarly,

$$E\{\mathbf{e}_i \mathbf{v}_{i-k}^T\} = -[\mathbf{A}(\mathbf{I} - \hat{\mathbf{K}}_0\mathbf{C})]^{k-1}\mathbf{A}\hat{\mathbf{K}}_0\mathbf{R} \qquad 5.24b$$

since $\mathbf{u}_k$ and $\mathbf{v}_k$ are uncorrelated white noise sequences.
Substituting (eqn. 5.24a-b) in (eqn. 5.21),

$$\Gamma_k = \mathbf{C}[\mathbf{A}(\mathbf{I} - \hat{\mathbf{K}}_0\mathbf{C})]^{k-1}\mathbf{A}[\mathbf{M}_1\mathbf{C}^T - \hat{\mathbf{K}}_0\Gamma_0) \qquad 5.25$$

For an optimal filter,

$$\hat{\mathbf{K}} = \mathbf{MC}^T(\mathbf{CMC}^T + \mathbf{R})^{-1} \qquad 5.26$$

so that Γ_k disappears for $k > 0$. When there are uncertainties in $\mathbf{Q}$ and $\mathbf{R}$, the calculated covariance $\mathbf{M}_c$ will be different from the true covariance $\mathbf{M}_1$ and the filter gain $\hat{\mathbf{K}}_0$ will be suboptimal. Therefore, Γ_k will not be zero and the optimal gain $\hat{\mathbf{K}}$ and the covariances $\mathbf{Q}$ and $\mathbf{R}$ can be obtained as follows:

(1) Obtain $\mathbf{M}_1\mathbf{C}^T$ by solving (eqn. 5.25) for $k = 1, \cdots, n$.

$$\mathbf{M}_1\mathbf{C}^T = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T \begin{vmatrix} \Gamma_1 + \mathbf{C}\mathbf{A}\hat{\mathbf{K}}_0\Gamma_0 \\ \Gamma_2 + \mathbf{C}\mathbf{A}\hat{\mathbf{K}}_0\Gamma_1 + \mathbf{C}\mathbf{A}^2\hat{\mathbf{K}}_0\Gamma_0 \\ \vdots \\ \Gamma_n + \mathbf{C}\mathbf{A}\hat{\mathbf{K}}_0\Gamma_{n-1} + \cdots \mathbf{C}\mathbf{A}^k\hat{\mathbf{K}}_0\Gamma_0 \end{vmatrix} \qquad 5.27$$

where $\mathbf{H}$ is defined in (eqn. 5.9).

(2) Calculate $\mathbf{R}$ using Γ_0 and $\mathbf{M}_1\mathbf{C}^T$.

$$\mathbf{R} = \Gamma_0 - \mathbf{C}(\mathbf{M}_1\mathbf{C}^T) \qquad 5.28$$

(3) Define $\mathbf{M}$ to be the error covariance associated with optimal gain matrix $\hat{\mathbf{K}}$:

$$\mathbf{M} = \mathbf{A}(\mathbf{M} - \hat{\mathbf{K}}\mathbf{C}\mathbf{M})\mathbf{A}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T \qquad 5.29$$

An equation for $\mathbf{M}_1$ is derived from (eqn. 5.17) :

$$\mathbf{M}_1 = \mathbf{A}[\mathbf{M}_1 - \hat{\mathbf{K}}_0\mathbf{C}\mathbf{M}_1 - \mathbf{M}_1\mathbf{C}^T\hat{\mathbf{K}}_0^T + \hat{\mathbf{K}}_0(\mathbf{C}\mathbf{M}_1\mathbf{C}^T + \mathbf{R})\hat{\mathbf{K}}_0^T]\mathbf{A}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T \qquad 5.30$$

subtracting (eqn. 5.30) from (eqn. 5.29) and defining $\delta\mathbf{M} = \mathbf{M} - \mathbf{M}_1$:

$$\begin{aligned}\delta\mathbf{M} = \mathbf{A}[&\delta\mathbf{M} - (\mathbf{M}_1\mathbf{C}^T + \delta\mathbf{M}\mathbf{C}^T)(\mathbf{C}_0 + \mathbf{C}\delta\mathbf{M}\mathbf{C}^T)^{-1}(\mathbf{C}\mathbf{M}_1 + \mathbf{C}\delta\mathbf{M}) \\ &+ \hat{\mathbf{K}}_0\mathbf{C}\mathbf{M}_1 + \mathbf{M}_1\mathbf{C}^T\hat{\mathbf{K}}_0^T - \hat{\mathbf{K}}_0\Gamma_0\hat{\mathbf{K}}_0^T]\mathbf{A}^T\end{aligned} \qquad 5.31$$

The optimal gain $\hat{\mathbf{K}}$ can using (eqn. 5.26) be written as

$$\hat{\mathbf{K}} = (\mathbf{M}_1\mathbf{C}^T + \delta\mathbf{M}\mathbf{C}^T)(\Gamma_0 + \mathbf{C}\delta\mathbf{M}\mathbf{C}^T)^{-1} \qquad 5.32$$

Steps (2) and (3) can be repeated until $|\delta\mathbf{M}_i|$ or $|\hat{\mathbf{K}}_i - \hat{\mathbf{K}}_{i-1}|$ becomes small compared to $|\mathbf{M}_i|$ or $|\hat{\mathbf{K}}_i|$, where $|.|$ denotes a suitable matrix norm. Another procedure would be to filter the available data $\mathbf{y}_i$ using the new $\hat{\mathbf{K}}$ and to repeat the whole procedure.

Solving (eqn. 5.31) for $\delta\mathbf{M}$ by using $\mathbf{M}_1\mathbf{C}^T$ from (eqn. 5.27), the optimal gain $\hat{\mathbf{K}}$ can be obtained from (eqn. 5.32). In actual practice, if a batch of observations $\mathbf{y}_i$ is available, these calculations might be repeated to improve the estimates. The innovation sequence $\mathbf{r}_i$ will become more white each iteration, resulting in better estimates for the autocorrelations $\Gamma_0, \Gamma_1, \cdots$, and therefore for $\mathbf{R}$ and $\mathbf{K}$. The estimate for $\mathbf{Q}$ can be obtained from (eqn. 5.30) but is only then uniquely defined if the dimension of the state noise vector r is equal or smaller than the dimension of the state vector n. Also for this method it can be proved that the estimates are asymptotically unbiased.

The efficiency of the correlation methods may be improved by using higher order correlations, i.e., $k > n$ to estimate $\mathbf{P}\mathbf{C}^T$ or $\mathbf{M}_1\mathbf{C}^T$. Moreover, the two methods can be combined by using the output correlation method as a start-up procedure since it does

not require any *a priori* estimates of **Q** and **R** or $\hat{\mathbf{K}}$. This method is discussed in [MEH70].

(Eqn. 5.31) requires the solution of a Lyapunov-type matrix equation for every iteration. A method that only requires matrix multiplications and is more stable than the here presented method is developed by Carew c.s. [CAR73]. For convenience, we only present the final results of the algorithm.

We define the following covariance matrix:

$$\mathbf{X}' = E\{(\hat{\mathbf{x}}_{i+1} - \mathbf{x}'_{i+1/i})(\hat{\mathbf{x}}_{i+1} - \mathbf{x}'_{i+1/i})^T\} \tag{5.33}$$

Where $\hat{\mathbf{x}}_i$ is the optimal linear estimate of $\mathbf{x}_i$ and $\mathbf{x}'_{i/i-1}$ the suboptimal estimate of $\hat{\mathbf{x}}_{i/i-1}$. For the optimal filter $\mathbf{X}' = 0$ and $\mathbf{K}_m$, the suboptimal Kalman gain matrix converges to **K**.

and we define **W** to be :

$$\mathbf{W} = \mathbf{CPC}^T + \mathbf{R} \tag{5.34}$$

the covariance matrix of the innovation sequence

Initially, **Q** and **R** are assigned the values $\mathbf{Q}_0$ and $\mathbf{R}_0$. Using these values a Kalman filter is designed from which the suboptimal steady-state filter gain $\mathbf{K}_0$ is obtained. The first step is to obtain estimates for the autocorrelation functions of the innovation sequence of the suboptimal filter $\hat{\mathbf{C}}_k$, $k = (0,1,2,\ldots,n)$. Using $\mathbf{K}_0$ and the autocorrelation functions, estimates **X**,**W** and **K** are obtained using the following formulas:

$$\mathbf{W}_m = \mathbf{C}_0 - \mathbf{CX}_m\mathbf{C}^T \tag{5.35}$$

$$\mathbf{K}_m = (\mathbf{F}\dagger\mathbf{G} - \mathbf{AX}_m\mathbf{C}^T)\mathbf{W}_m^{-1} \tag{5.36}$$

$$\mathbf{X}_{m+1} = (\mathbf{A} - \mathbf{K}_0\mathbf{C})\mathbf{X}_m(\mathbf{A} - \mathbf{K}_0\mathbf{C})^T + (\mathbf{K}_0 - \mathbf{K}_m)\mathbf{W}_m(\mathbf{K}_0 - \mathbf{K}_m)^T \tag{5.37}$$

with m an iteration index and $\mathbf{F}\dagger$ is defined by:

$$\mathbf{F}\dagger = (\mathbf{F}^T\mathbf{F})^{-1}\mathbf{F}^T$$

the generalized inverse of **F**, and **F** is the system observability matrix of rank n.

$$\mathbf{F}^T = \left[\mathbf{C}^T \mid \mathbf{A}^T\mathbf{C}^T \mid \cdots \mid (\mathbf{A}^T)^{n-1}\mathbf{C}^T\right]$$

and **G** by :

$$\mathbf{G} = \begin{bmatrix} \mathbf{C}_1 + \mathbf{CK}_0\mathbf{C}_0 \\ \mathbf{C}_2 + \mathbf{CK}_0\mathbf{C}_1 + \mathbf{CAK}_0\mathbf{C}_0 \\ \vdots \\ \mathbf{C}_n + \mathbf{CK}_0\mathbf{C}_{n-1} + \cdots + \mathbf{CA}^{n-1}\mathbf{K}_0\mathbf{C}_0 \end{bmatrix}$$

A simple choice for $\mathbf{X}_0$ is the null matrix. This is further justified by the fact that $\mathbf{X}^{\cdot} = 0$ if $\mathbf{K}_0$ happens to be the optimal Kalman gain matrix. The iterative scheme (eqn. 5.35-37) converges uniquely to the optimal gain $\mathbf{K}$ if the autocorrelation functions are accurate. However, the finiteness of the sample size and other experimental and round-off errors, introduce errors in the autocorrelations. This sets a limit on the accuracy with which $\mathbf{K}$ can be determined.

5.5. COVARIANCE MATCHING TECHNIQUES

The basic idea behind the covariance matching techniques is to make the residuals consistent with their theoretical covariances. For example, consider the innovation sequence $\mathbf{r}_i$, which has a theoretical covariance of $(\mathbf{C}\mathbf{P}_{i/i-1}\mathbf{C}^T + \mathbf{R})$. If it is noticed that the actual covariance is much larger than the theoretical value obtained from the Kalman filter, then the process noise $\mathbf{Q}$ should be increased. This has the effect of increasing $\mathbf{P}_{i/i-1}$ and bringing the actual covariance of $\mathbf{r}_i$ closer to the theoretical value.

One method that adaptively estimates the *a priori* statistics is given by Myers [MYE76]. He also uses the innovation sequence to adapt the noise covariance matrices.

$$\mathbf{r}_k = \mathbf{y}_k - \mathbf{C}\,\mathbf{x}_{k/k-1} \tag{5.38}$$

Consider a sample dataset consisting of N measurements. The covariance matrix of $\mathbf{r}_k$ is given by:

$$Cov(\mathbf{r}_k) = E[(\mathbf{y}_k - \mathbf{C}\,\mathbf{x}_{k/k-1})(\mathbf{y}_k - \mathbf{C}\,\mathbf{x}_{k/k-1})^T] \tag{5.39}$$

An unbiased estimator for the covariance matrix of $\mathbf{r}_k$ is :

$$Cov(\mathbf{r}_k) = \frac{1}{N-1}\sum_{j=1}^{N} r_j r_j^T \tag{5.40}$$

It can be shown that the expected value of this quantity is :

$$Cov(\mathbf{r}_k) = \mathbf{C}\,\mathbf{P}^{\infty}\mathbf{C}^T + \mathbf{R} \tag{5.41}$$

The following estimate of $\hat{\mathbf{R}}$ can be obtained by using (eqn. 5.40) and (eqn. 5.41) :

$$\hat{\mathbf{R}} = \frac{1}{N-1}\sum_{j=1}^{N} \mathbf{r}_k\,\mathbf{r}_k^T - \mathbf{C}\mathbf{P}^{\infty}\mathbf{C}^T \tag{5.42}$$

$\mathbf{P}^{\infty}$ is the *a priori* covariance matrix of the state. This means that it is the steady state value of $\mathbf{P}_{k+1/k}$. (Eqn. 5.42) is of the same type as (eqn. 5.28) except that $\mathbf{P}^{\infty}$ is obtained from the Kalman filter. In certain applications where $\mathbf{Q}$ is known, this has shown to be a good approximation. However convergence of covariance matching techniques is doubtful.

The same can be done for the noise covariance matrix **Q**. The state relation is :

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{G}\mathbf{w}_k \tag{5.43}$$

The true states $\mathbf{x}_{k+1}$ and $\mathbf{x}_k$ are unknown, so $\mathbf{w}_k$ can not be determined, but an approximation for $\mathbf{w}_k$ is :

$$\mathbf{q}_{k+1} = \hat{\mathbf{x}}_{k+1/k} - \mathbf{A}\hat{\mathbf{x}}_{k/k-1} \tag{5.44}$$

where $\mathbf{q}_k$ is defined as the state noise sample at instant k. If $\mathbf{q}_k$ is assumed to be representative of $\mathbf{G}\mathbf{w}_k$ then it may be considered independent and identically distributed. An unbiased estimator for the noise covariance matrix $\mathbf{G}\mathbf{Q}\mathbf{G}^T$ is :

$$Cov(\mathbf{q}) = \frac{1}{N-1}\sum_{j=1}^{N}\mathbf{q}_j\mathbf{q}_j^T \tag{5.45}$$

Another estimator for the covariance matrix is:

$$Cov(\mathbf{q}) = \mathbf{A}[\mathbf{P}^{\infty} - \hat{\mathbf{K}}^{\infty}\mathbf{C}\,\mathbf{P}^{\infty}]\mathbf{A}^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T \tag{5.46}$$

again an estimator for the noise covariance matrix $\hat{\mathbf{G}}\hat{\mathbf{Q}}\hat{\mathbf{G}}^T$ can be found by:

$$\hat{\mathbf{G}}\hat{\mathbf{Q}}\hat{\mathbf{G}}^T = \frac{1}{N-1}\sum_{j=1}^{N}\mathbf{q}_j\mathbf{q}_j^T - \mathbf{A}[\mathbf{P}^{\infty} - \mathbf{K}^{\infty}\mathbf{C}\mathbf{P}^{\infty}]\mathbf{A}^T \tag{5.47}$$

(Eqn. 5.47) does not give a unique solution for **Q** if **G** is of rank less than n. (Eqn. 5.41) is an approximation since **P** does not represent the actual covariance since the actual values of **Q** and **R** are unknown.

5.6. SOME CONSIDERATIONS

It is obvious that the noise covariance matrices $\hat{\mathbf{R}}$ and $\hat{\mathbf{Q}}$ may become negative definite in numerical applications, especially when a small amount of data has been processed. Therefore, the diagonal elements of $\hat{\mathbf{R}}$ and $\hat{\mathbf{Q}}$ are always reset to the absolute value of their estimates. During filter initialization, the noise matrices are poor indicators of the noise, especially since only the steady state values $\mathbf{K}^{\infty}$ and $\mathbf{P}^{\infty}$ are used. This requires a fading memory approach in which successive samples are multiplied by a weight factor given by :

$$w_k = (k-1)(k-2)...(k-\beta)/k^{\beta} \tag{5.48}$$

which has the property that it converges to 1 as k approaches ∞ and that the use of invalid noise samples is delayed for the first β stages.
Several other methods are proposed but convergence of covariance matching techniques has never been proved. In this case it is also not always possible to find **Q** unless the number of unknowns is restricted so that $r \leqslant n$ e.g., the dimension of the state vector is equal or larger than the dimension of the state-noise vector.

The innovation correlation methods seem to be the most promising methods for adapting the Kalman filter since the autocorrelation have already been calculated to test the optimality for the Kalman filter. And since the method developed by Carew (eqn. 5.35-37) involves no Lyapunov type of equations, this one has a better numerical behaviour than the method developed by Mehra (eqn. 5.27-5.29) [CAR73]. Output correlation methods are restricted to stationary outputs or a stable **A** but do not require initial estimates for **Q**, **R** and **K** and could be considered as start-up methods. Covariance methods seem simple and straightforward but since they use incorrect state error covariance matrices their convergence is doubtful. Numerical problems can be prevented by using square root formulations.

CHAPTER 6

OPTIMAL CONTROL

6.1. WHY OPTIMAL CONTROL?

For single-input, single output (SISO) systems it is possible to design a compensator which places the closed-loop poles wherever the designer wants them to be. Since the closed-loop poles determine the speed (bandwidth) and the damping of the response, why should we want to develop new mathematic tools? There are several good reasons for that [FRI86].

The first reason for seeking an optimal controller is that for multi-input, multi-output (MIMO) systems, the pole placement technique does not completely specify the controller gains. When we have a nth-order system with m inputs and all states accessible, a controller has, in the case of state feedback, nm parameters to determine, but only n closed-loop pole locations. This means that there are an infinite number of ways to obtain the same closed-loop pole locations. From a practical viewpoint, the fact that there are more adjustable parameters than needed to achieve the desired closed- loop pole location, is a great opportunity since other things can be accomplished besides placing the closed-loop poles but which one is the best? By choosing a control law to optimize a, yet to be specified, performance criterion, the designer can choose one possibility out of an infinite number.

Another reason for seeking an optimal controller is that the designer may not know the desirable closed-loop pole locations. Choosing poles far from the origin, in the s-domain, may give very fast responses but require control signals that are too large too be produced by the available power source. Too large input signals will lead to saturation and a behaviour of the system that can not be modeled by the same linear model. To avoid these problems it is often necessary to limit the speed of response to that which can be achieved without saturation of the input signals. Another reason is that high-gain systems are typically accompagnied by noise problems that the designer may want to avoid.

6.2. FORMULATION OF THE OPTIMAL CONTROL PROBLEM

The object of optimal control is to specify an input vector $\mathbf{u}_k$ which drives a system to a specified target state in such a way that, during the process, a defined 'cost function' is minimized.

6.2.1. LINEAR OPTIMAL QUADRATIC GAUSSIAN CONTROL

Optimal control is in no way limited to linear systems but when a linear quadratic cost function is specified for a time independent linear system, optimal control strategy results in a particular case of pole shifting by state feedback. If the system is corrupted by Gaussian white noise processes, the control routine is called LQG (Linear Quadratic Gaussian) [FRI86].

Consider the following time independent system:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{G}\mathbf{w}_k \tag{6.1}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{v}_k \tag{6.2}$$

A linear quadratic cost function is defined:

$$J = E\{\sum_{k=0}^{\infty} [\mathbf{x}_k^T \mathbf{E}\mathbf{x}_k + 2\mathbf{u}_k^T \mathbf{G}\mathbf{x}_k + \mathbf{u}_k^T \mathbf{F}\mathbf{u}_k]\} \tag{6.3}$$

where **E** is a positive semidefinite symmetric (n-by-n) matrix and **F** a positive definite symmetric (m-by-m) matrix. **G** is a positive semidefinite (m-by-n) matrix which is normally set to zero unless we want to minimize the output as follows:

$$J = E\{\sum_{k=0}^{\infty} \mathbf{y}_k^T \mathbf{y}_k\} \tag{6.4}$$

In this case $\mathbf{E} = \mathbf{C}^T\mathbf{C}$, $\mathbf{F} = \mathbf{D}^T\mathbf{D}$ and $\mathbf{G} = \mathbf{D}^T\mathbf{C}$. with $\mathbf{y}_k$ given by (eqn. 6.2).

The **E** matrix can be used to specify a trajectory the state vector has to follow and is often called *state weighting* matrix. The matrix **F** can be used to put limits on the input vector $\mathbf{u}_k$ and is often called *control weighting* matrix.

The problem is to find an input $\mathbf{u}_k$ which drives the state from $\mathbf{x}_0$ to $\mathbf{x}_\infty$ with minimal J. The solution to this problem is found to be :

$$\mathbf{u}_k = \mathbf{L}\hat{\mathbf{x}}_{k/k-1} \tag{6.5}$$

where

$$\mathbf{L} = -(\mathbf{B}^T\mathbf{P}\mathbf{B} + \mathbf{F})^{-1}(\mathbf{A}^T\mathbf{P}\mathbf{B} + \mathbf{G}^T)^T \tag{6.6}$$

and **P** is the (n-by-n) matrix solution of the algebraic Riccati equation:

$$\mathbf{P} = \mathbf{A}^T\mathbf{P}\mathbf{A} - (\mathbf{A}^T\mathbf{P}\mathbf{B} + \mathbf{G}^T)(\mathbf{B}^T\mathbf{P}\mathbf{B} + \mathbf{F})^{-1}(\mathbf{A}^T\mathbf{P}\mathbf{B} + \mathbf{G}^T)^T + \mathbf{E} \tag{6.7}$$

with $\hat{\mathbf{x}}_{k/k-1}$ the optimal prediction of the state obtained by the Kalman filter. The power of the LQG problem lies in the fact that it can be divided into two separate problems: obtaining an optimal prediction of the state of a stochastic system using a Kalman filter, and obtaining an optimal feedback for a deterministic system with a predicted state (separation principle) [KWA72]. Figure 6.1 shows how the separation

principle divides the LQG-algorithm into two separate problems.

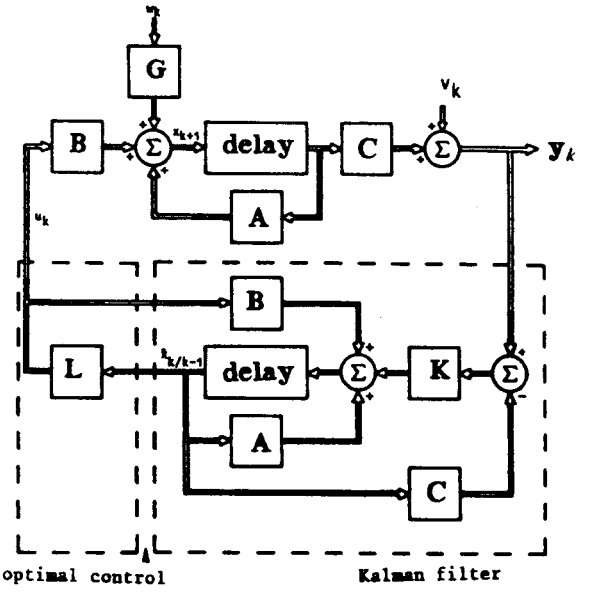

Fig. 6.1
The separation principle.

The fact that the required trajectory must be expressed in terms of $\mathbf{x}^T\mathbf{E}\mathbf{x}$ places considerable limitations on the trajectories which can be specified. Furthermore, the mere fact that a trajectory can be specified this way is no guarantee that the optimal control can achieve it, the cost function J is merely minimized, not necessarily brought to zero.

Furthermore, the designer is left with the selection of the weighting matrices **E** and **F**, but often minimization of (eqn 6.3) is not the true design objective. The problem is that the true design objective often cannot be expressed in mathematical terms. Formulating the design objective in the form of a quadratic sum is a practical compromise between formulating a real problem that cannot be solved and formulating a maybe somewhat artificial problem that can be solved easily.

Since the relationship between the weighting matrices **E** and **F** and the dynamic behaviour of the closed-loop system are quite complex, it is impossible to predict the effect on closed-loop behaviour of a given pair of weighting matrices. A suitable approach for a designer would be to solve for a range of **E** and **F** matrices and to calculate the corresponding closed-loop behaviour. With the here presented software this would take a few hours of time, depending on the number of inputs and outputs and states.

6.2.2. MINIMUM VARIANCE CONTROL STRATEGY

It can be shown that the LQG approach leads to a minimum variance control strategy [GOO84]. The importance of minimum variance control is that by reducing the variance of a given variable, the set point, y^*, can be put at a higher value while

still ensuring that a certain proportion of the output meets a given acceptance criterion. This can lead to greater throughput or reduced costs. This is illustrated in figure 6.2.

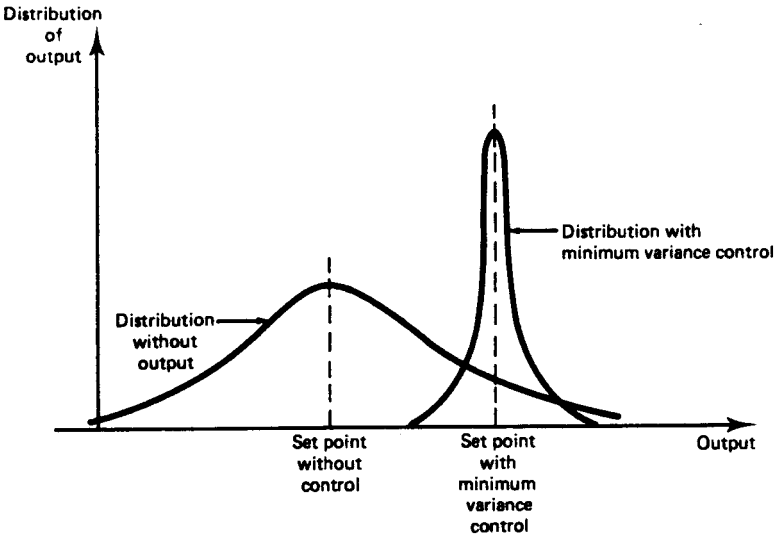

Fig. 6.2
The importance of minimum variance controllers

6.3. THE ALGORITHM

The algorithm used to solve the algebraic Riccati equation, has been derived by P. van Dooren [DOO81]. The algorithm basically solves a generalized eigenvalue problem:

$$\mathbf{Mx} = \lambda \mathbf{Nx} \tag{6.8}$$

where $\mathbf{N}$ is not necessarily invertible but where the pencil $\lambda\mathbf{N} - \mathbf{M}$ is regular. i.e.,

$$\det(\lambda\mathbf{N} - \mathbf{M}) \not\equiv 0 \tag{6.9}$$

It can be shown that solving (eqn. 6.7) (with $\mathbf{G} = 0$) is equivalent to solving for the "stable" deflating subspace of the pencil:

$$\lambda \begin{bmatrix} \mathbf{I} & \mathbf{B}\mathbf{F}^{-1}\mathbf{B}^T \\ 0 & \mathbf{A}^T \end{bmatrix} - \begin{bmatrix} \mathbf{A} & 0 \\ -\mathbf{E} & \mathbf{I} \end{bmatrix} \tag{6.10}$$

where the stable eigenvalues are those inside the unit circle. The "stable" deflating subspace is the subspace spanned by the eigenvectors of (eqn. 6.10) that correspond to stable eigenvalues. (Eqn. 6.10) is better known as the Euler-Lagrange equation [DAM86], [KWA72]. The solution is found by obtaining the stable eigenvalues and eigenvectors.

To avoid matrix inversions, (eqn. 6.10) can be rewritten as [DOO81]:

$$\lambda \begin{vmatrix} \mathbf{I} & 0 & 0 \\ 0 & \mathbf{A}^T & 0 \\ 0 & \mathbf{B}^T & 0 \end{vmatrix} - \begin{vmatrix} \mathbf{A} & 0 & -\mathbf{B} \\ -\mathbf{E} & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{F} \end{vmatrix} \qquad 6.11$$

The algorithm has 5 stages:

1. Orthogonal row transformations will transform the $(2n+p)$-by-$(2n+p)$ matrices of (eqn. 6.11) into $2n$-by-$2n$ matrices given by (eqn. 6.10).
2. **M** is reduced to upper Hessenberg form by a Householder reduction and at the same time **N** is transformed to upper triangular form.
3. A double shift QR algorithm is used to reduce **M** to quasi-triangular form with no two consecutive subdiagonal elements nonzero while maintaining the triangular form of **N**. The eigenvalues are extracted and reordered into stable and unstable ones.
4. The n-by-$2n$ eigenvectors are obtained from **M** and **N** and are transformed back to the original coordinate system to form the following stable basis for the deflating pencil:

$$\begin{vmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{vmatrix} \qquad 6.12$$

5. The n-by-n **P** can be obtained from (eqn. 6.12) :

$$\mathbf{P} = \mathbf{X}_2\mathbf{X}_1^{-1} \qquad 6.13$$

This method is better known as the solution by diagonalization as described in [DAM83]. The actual algorithm is a numerical stable implementation of this method [DOO81], [MOL73]. A small change in the matrices of (eqn. 6.11) makes it possible to solve also for $\mathbf{G} \neq 0$. The controller provides an asymptotically stable closed-loop system if **E** is posive semidefinite, **F** positive definite and [KWA72]:

(1) (**A**,**B**) either completely controllable or stabilizable by state feedback (necessary condition).

(2) (**A**,**V**) completely observable with **V** any n-by-n matrix that satisfies $\mathbf{V}\mathbf{V}^T = \mathbf{E}$ (sufficient condition).

6.4. THE EFFECT OF INACCURATE NOISE MATRICES

A simple, scalar problem will be used to show the effect of using the right values for the noise matrices **Q** and **R** in the Kalman filter.

Consider the following system:

$$x_{k+1} = 0.8\, x_k + 0.5\, u_k + w_k \qquad 6.13a$$

$$y_k = x_k + v_k \qquad 6.13b$$

with:

$$E\{w_k\} = E\{v_k\} = 0$$
$$E\{w_k^2\} = E\{v_k^2\} = 1$$

and

$$E\{x_0\} = 10$$
$$E\{(x_0 - E\{x_0\})^2 = 0.1$$

If the functional to be minimized is:

$$J = E\{\tfrac{1}{2} \sum_{k=0}^{99} (x_k^2 + u_k^2)\}$$

the resulting value of J is 405.

If however the erroneous value $E\{w_k^2\} = 0.1$ is used, the resulting functional obtains the value 412 [OLC83].
A comparison of these two values shows the degradation of the sub-optimal filter.

CHAPTER 7

DISCUSSION OF THE RESULTS

7.1. INTRODUCTION

PRIMAL (Package for Real time Interactive Modeling Analysis and Learning) is a software package developed at the Physics department, section Control and Measurement, of the Technical Universtity of Eindhoven. This package enables a user to obtain, interactively, "raw" data from a process. PRIMAL presents the user with modules for data conditioning, model estimation, model validation and model simulation. But the lack of control modules was considered a serious problem by the users of PRIMAL. Therefore a LQG-module has been implemented, although the work has been focused on state estimation using a Kalman filter. For this reason, four PRIMAL modules have been written that can be used to design and test a Kalman filter for a stochastic state space model. Together with an optimal feedback matrix, calculated by a fifth module, they implement an optimal control strategy.

The first module, KALSIM [MEU88a], simulates a stochastic state space model of the form (5.1-2) with specified $\mathbf{Q}$, $\mathbf{R}$ and initial $\mathbf{x}_0$, $\mathbf{P}_0$.

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{G}\mathbf{w}_k$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{v}_k$$

This module has been developed because SIMSYS (a state-space simulator in PRIMAL) does not provide means to simulate a state-space system with specified noise covariance matrices for the state and output vector.
The module simulates the noise corrupted state and output vectors as a function of time with their mean and noise covariance matrices. To test if the simulation is consistent with the specified variances, the 95 % limits for the observations of the state and output vector are calculated as well as the amount that is "out of bound", which should not exceed the limit of 5 % [MEU88a]. If the number of "out of bound" exceeds 5 %, the user may specify different seeds for the random number generator and rerun the simulation. KALSIM produces an output dataset which contains the simulated state vector and simulated output vector, together with their 95 % limits. As an example, figures 7.1a-b show the simulated vectors and their bounds for a SISO system.

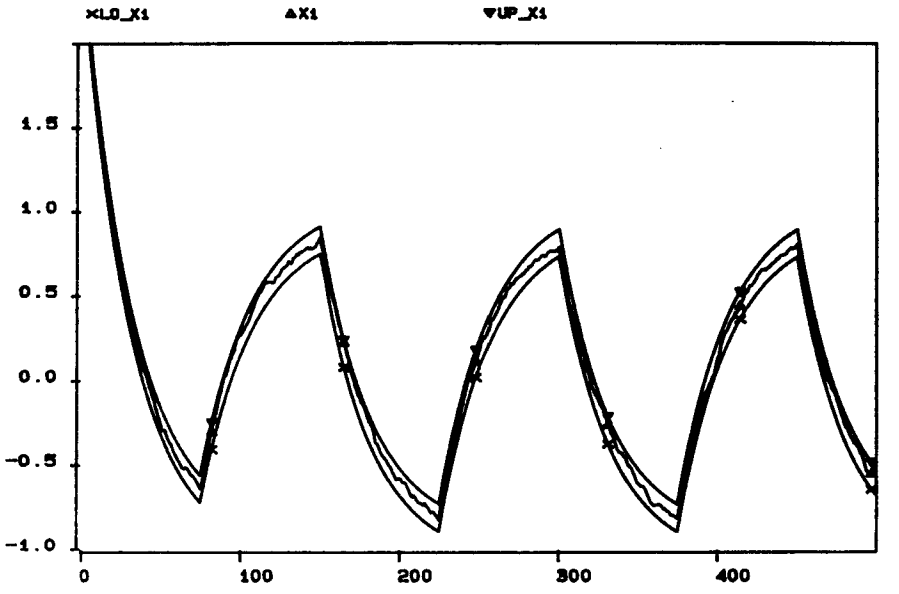

Fig. 7.1a
The simulated state vector and the 95 % bounds as function of the time index k.

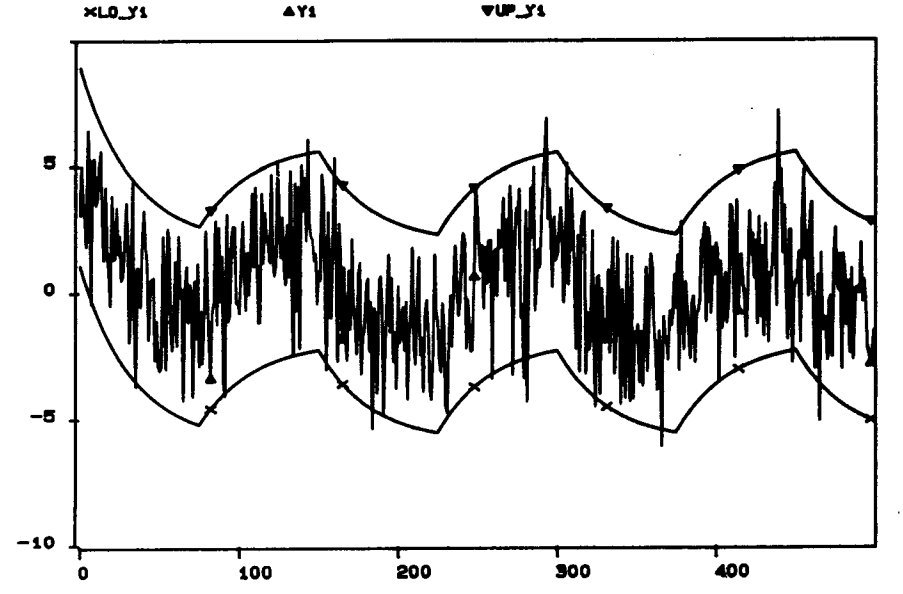

Fig. 7.1b
The simulated output vector and the 95 % bounds as function of the time index k.

The second module is the Kalman filter KALMAN [MEU88b] which calculates the steady-state Kalman filter, given a linear stochastic state space model and the state and measurement noise covariance matrices **Q** and **R**. It calculates the steady-state covariance matrix **P**, the steady-state Kalman gain matrix **K** and the covariance matrix of the innovation vector **W** using a square root algorithm. The calculations are repeated recursively until steady-state is reached and the module keeps track of the convergence history.

Figures 7.2a-b illustrate the convergence history of the state error covariance matrix **P** and the Kalman gain matrix **K** as produced by KALMAN for a MIMO system.

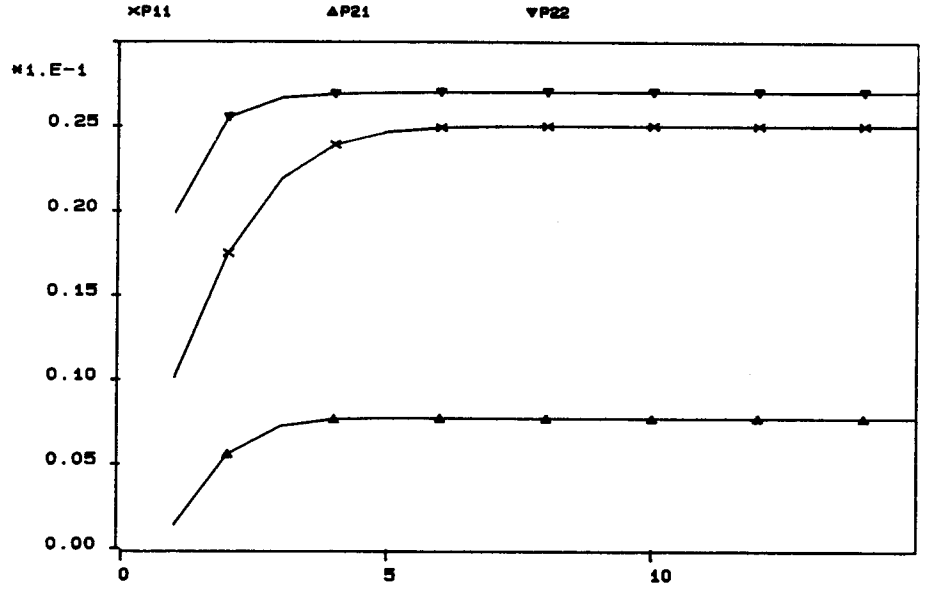

Fig. 7.2a
The state error covariance matrix.

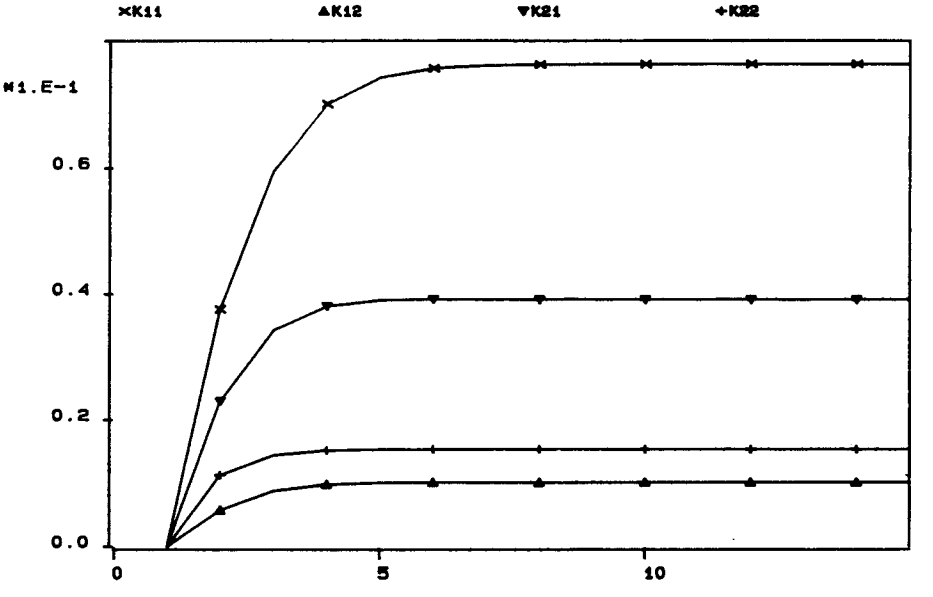

Fig. 7.2b
The Kalman gain matrix.

The third module KALEST [MEU88c] uses the computed Kalman filter and the simulated data to estimate the state vector. KALEST calculates the state vector and the innovation vector with their corresponding variances. The innovation covariance may be compared with the estimate computed by KALMAN. The module calculates the 95 % limits for the state and innovation vector and the number of observations outside these limits. As an example of the data presented to the user, figures 7.3a-b show the state vector, the

innovation vector and the 95 % limits.

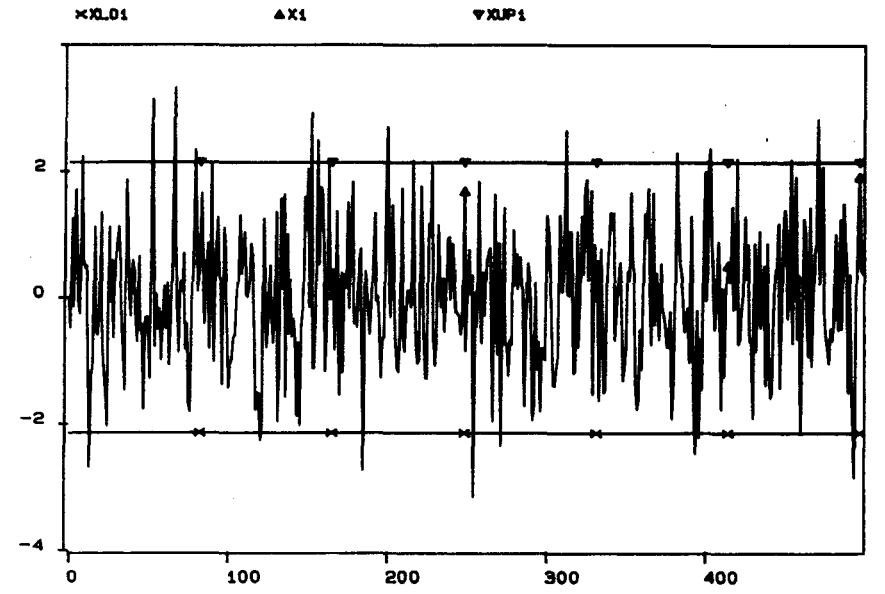

Fig. 7.3a
The state error vector with the 95 % bounds as function of the time index k.

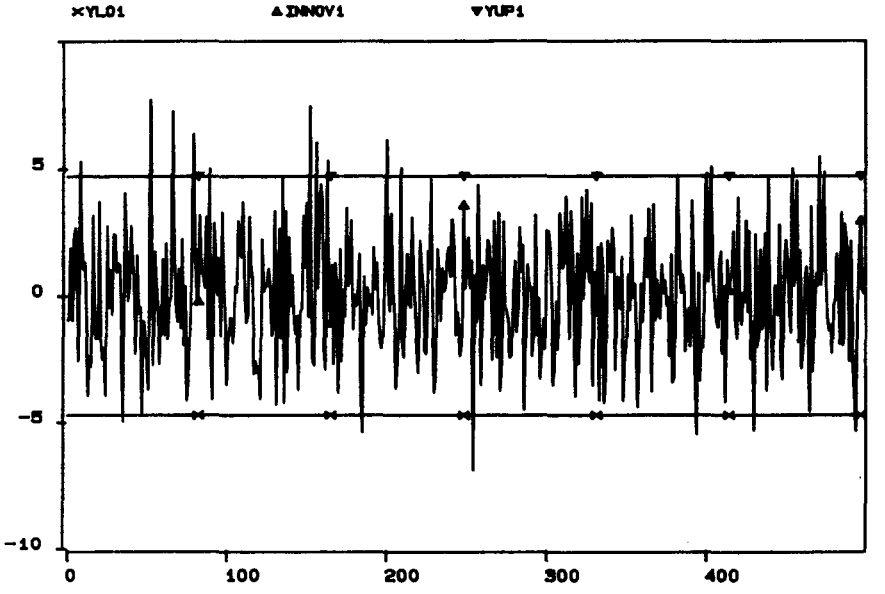

Fig. 7.3b
The innovation vector with the 95 % bounds as function of the time index k.

The 95 % bounds are calculated using the, by KALMAN, computed covariance matrices for the state error **P** and the innovations **W**:

$$-1.96\sqrt{\mathbf{P}_{ii}} < [\mathbf{x}_k - \mathbf{x}_k^l]_i < 1.96\sqrt{\mathbf{P}_{ii}}$$

with $\mathbf{x}'$ the "truth" vector (calculated by KALSIM).

$$-1.96\sqrt{\mathbf{W}_{ii}} < [\mathbf{r}_k]_i < 1.96\sqrt{\mathbf{W}_{ii}}$$

where k is the time index and i the i-th element of a vector and ii the i-th diagonal element of a matrix.
5 percent of the observations are allowed to lie outside these limits. This is only valid in case of Gaussian white noise.

The fourth module WTEST [MEU88d] is a whiteness test module. It tests if a sequence $\mathbf{r}_i$ is zero mean white noise by looking at the sample mean (should be less than a certain bound) and the correlation function (less than 5 % of the correlation values should lie outside the corresponding bounds). Figure 7.4 shows the normalized correlation coefficient ρ_k and the 95 % bounds as function of the lag k.

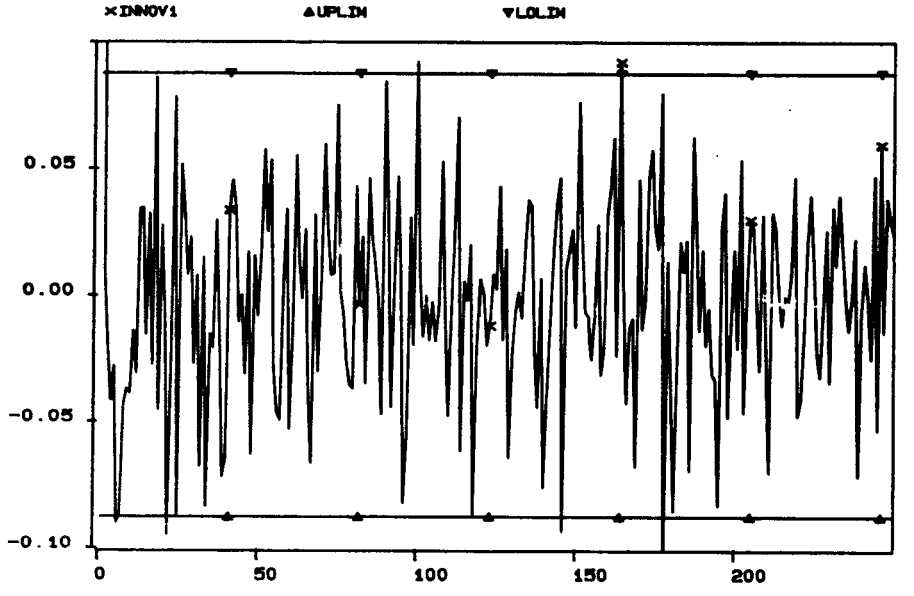

Fig. 7.4
The normalized correlation function and the 95 % bounds as function of the lag.

The 95 % confidence interval is constructed using the fact that the normalized correlation coefficient is normally distributed for a large number of lags and has zero mean and covariance $1/N$. Therefore,

$$-1.96/\sqrt{N} < \rho_k < 1.96/\sqrt{N}$$

where N is the number of samples used. Thus only 5 % of the normalized sample covariances are allowed to exceed this bound for the sequence to be called statistically white.
The bound for the sample mean is obtained by a statistical hypothesis test given by:

$$\mathbf{H0}: E\{\mathbf{r}_k\} = 0$$

$$\mathbf{H1}: E\{\mathbf{r}_k\} \neq 0$$

The test is implemented at the 0.05 significance level (95 %) with the threshold, called bound mean, calculated using the sample mean estimator for $E\{\mathbf{r}_k\}$:

$$1.96\ \sqrt{cov(\mathbf{r}_k)/N}$$

with N the number of samples and cov the covariance. The calculated mean should be smaller than the threshold.

Together these 4 routines (KALSIM, KALMAN, KALEST, WTEST) form the state estimation modules. They can be used to design a Kalman filter and test its performance.

The Kalman filter is functioning optimally when:

(1) The innovations are zero mean, white noise (calculated by WTEST).

(2) Less than 5 % of the observations for the innovations are "out of bound".

(3) The, by KALMAN, calculated variance of the innovation vector is equal to the, by KALEST, observed variance.

These 3 criteria will be used in the next sections to check the Kalman filter performance.

If the innovations are white noise and zero mean, but there is an inconsistency between the estimated and calculated variance of the innovations, the Kalman filter can still be functioning close to optimum, depending on the actual values for **Q** and **R**. Sections 7.4-6 will treat the influence of the value **Q/R** on the behaviour of the optimal filter and suboptimal filters.

A fifth module RIC has been developed that calculates the steady-state optimal feedback matrix **L** for a stochastic state space system. The module solves the steady-state Riccati equation using the in chapter 6 described algorithm [MEU88e]. This module forms, together with the Kalman filter, a LQG- algorithm.

7.2. A PRACTICAL EXAMPLE

In reality we often do not have exact knowledge about the system and the noise covariance matrices. All uncertainties will be condensed into **Q** and **R**. To show this we have used the following SISO model with a 1-dimensional state vector:

$$x_{k+1} = 0.97\, x_k + u_k + w_k \qquad 7.1a$$

$$y_k = 2\, x_k + v_k \qquad 7.1b$$

with $\mathbf{Q} = 1.e-3$ and $\mathbf{R} = 1.0$.

We have used the stochastic state space simulator KALSIM to simulate this system. Figure 7.5 shows the impulse respones of the simulated system.

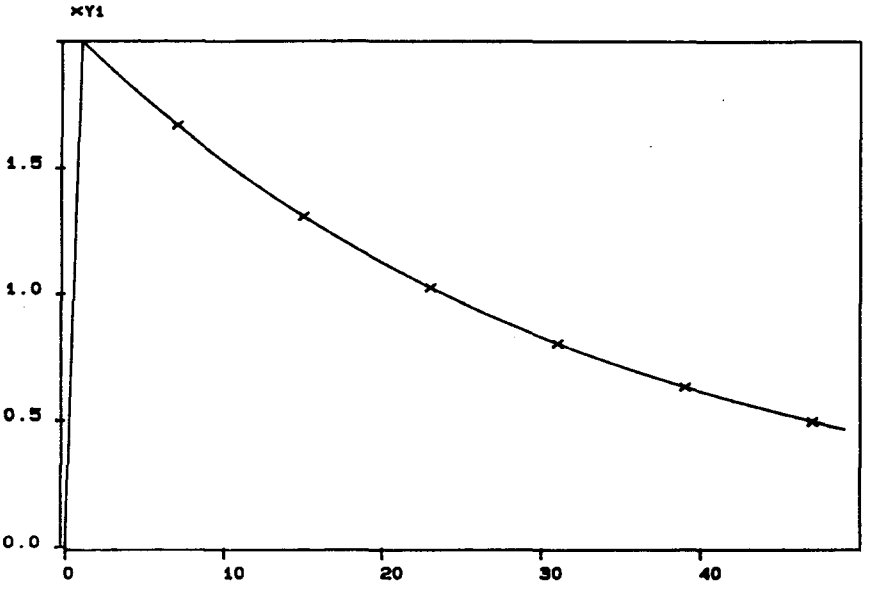

Fig. 7.5
The impulse response for the simulated system as function of the time-index k.

The calculated input and output vectors are used by MARKOV [LIN87] to estimate the impulse response of the system. HANKEL (a realization algorithm that converts the impulse response into a state space system) is used to obtain the following state space system [GER88]:

$$x_{k+1} = 0.9645\, x_k - 1.42\, u_k \qquad 7.2a$$

$$y_k = -1.41\, x_k \qquad 7.2b$$

by using a state vector transformation $\hat{x}_k = Tx_k$ and $T = -1/1.42$, we obtain the following, equivalent system:

$$\hat{x}_{k+1} = 0.9645\ \hat{x}_k + u_k \qquad 7.3a$$

$$y_k = 2.00\ y_k \qquad 7.3b$$

The by MARKOV calculated impulse response (M11) and the fit by HANKEL (MR11) are shown in figure 7.6 (note the difference with fig. 7.5. MR11 decreases faster than the original response):

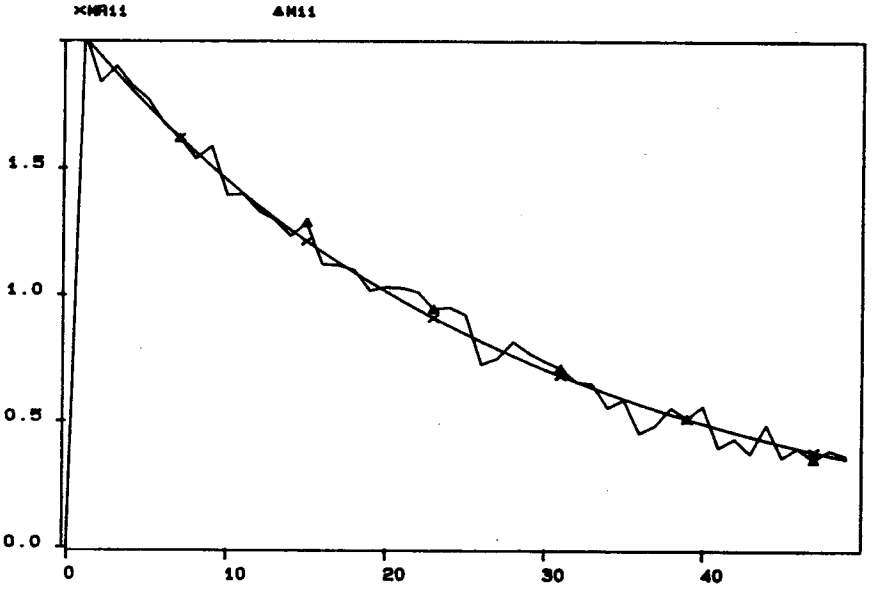

Fig. 7.6
The calculated and fitted impulse response as function of the time-index k.

Using the Kalman filter module KALMAN, a Kalman filter is designed for the state space model (eqn. 7.2a-b) and $\mathbf{Q} = 1.e-3$ and $\mathbf{R} = 1.$. After 104 iterations, the Kalman gain reaches steady-state.

The value of **P**, **K** and **W** are:

P	K	W
1.11e-2	-1.49e-2	1.02

Table 7.1
Results from KALMAN for the SISO-system (7.2).

The calculated Kalman gain is used by KALEST to estimate the state and innovation vector. The results for the innovation vector are presented in table 7.2:

out of bound	estimated variance	calculated variance
43=8.6%	1.02	1.25

Table 7.2
Results from KALEST for the SISO-system (7.2).

KALEST shows that more than 5 percent of the innovation vector is "out of bound" and the calculated variance is higher than the estimated one. The residuals clearly show a dynamic component (fig. 7.7):

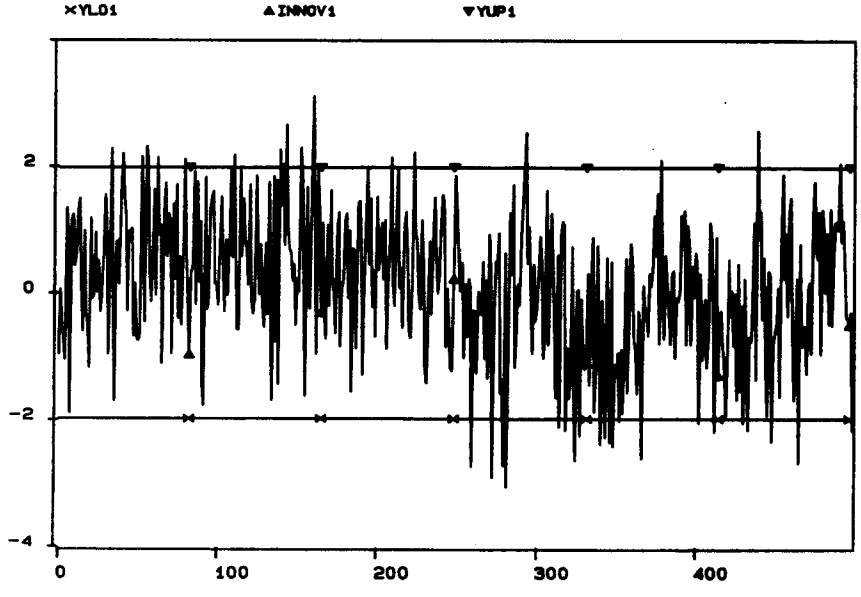

Fig. 7.7
The residuals and the 5 % limits as function of the time-instant k.

We use WTEST to see what might be the cause of this. The sample mean, the bound mean and the number of correlation values that are out of bound are presented in table 7.3:

sample mean	bound mean	out of bound
0.14	0.098	70=28.1%

Table 7.3
Results from WTEST for the SISO-system (7.2).

From table 7.3 it is obvious that the innovation vector is neither zero mean nor white noise. This can be explained better when we look at the correlation function (fig. 7.8). We

can clearly observe a coloured noise component in the correlation function. This is caused by the difference between the real system matrix (0.97) and the HANKEL estimated system matrix (0.9645).

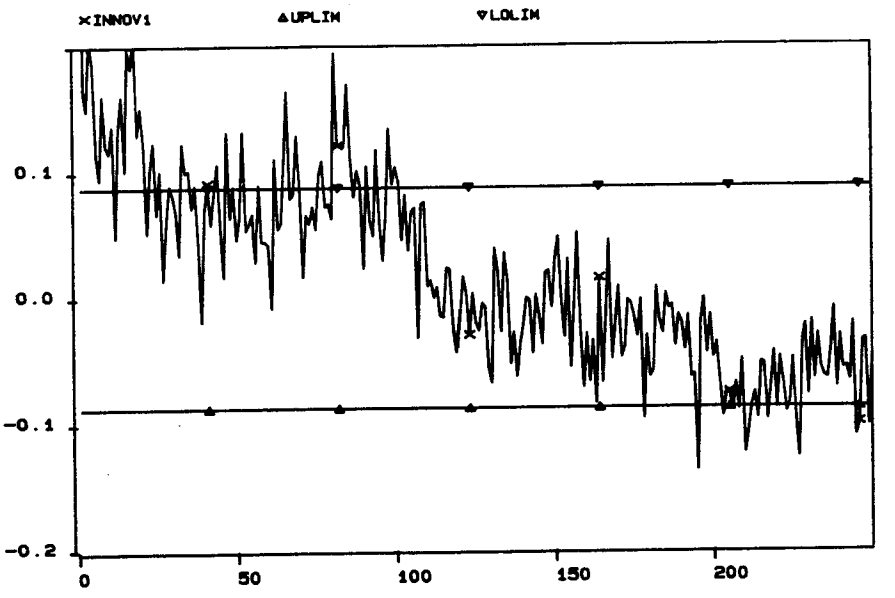

Fig. 7.8
The correlation function versus lag.

We will use the in chapter 5 developed techniques to adapt **Q** and **R**, using the values of the calculated correlation function, C_0 and C_1. The necessary values of the correlation function are found to be (table 7.4):

C_0	C_1	scale factor
1	0.168	1.25

Table 7.4
The first two values of the correlation function obtained from WTEST.

The values have been normalized by dividing them by the scale factor.

Using (eqn. 5.27-28) we obtain **Q** = $1.32e-2$ and **R** = 1.00. Compare these values with the original values **Q** = $1.e-3$ and **R** = 1.00. The extra uncertainty, introduced by MARKOV and HANKEL in the system matrix is added to the existing state noise.

We design a new Kalman filter using the adapted values (table 7.5):

P	K	W
7.13e-2	-8.47e-2	1.15

Table 7.5
Results from KALMAN for adapted SISO-system

To check if these values are optimal we use KALEST to estimate the state and innovation vector with the following results:

out of bound	estimated variance	calculated variance
4.4%	1.15	1.12

Table 7.6
The results from KALEST for the system with adapted noise matrices

From table 7.6 can be concluded that less than 5 percent is out of bound and the calculated variance is close to the estimated one. The innovation sequence looks better, the dynamic component has disappeared (fig. 7.9).

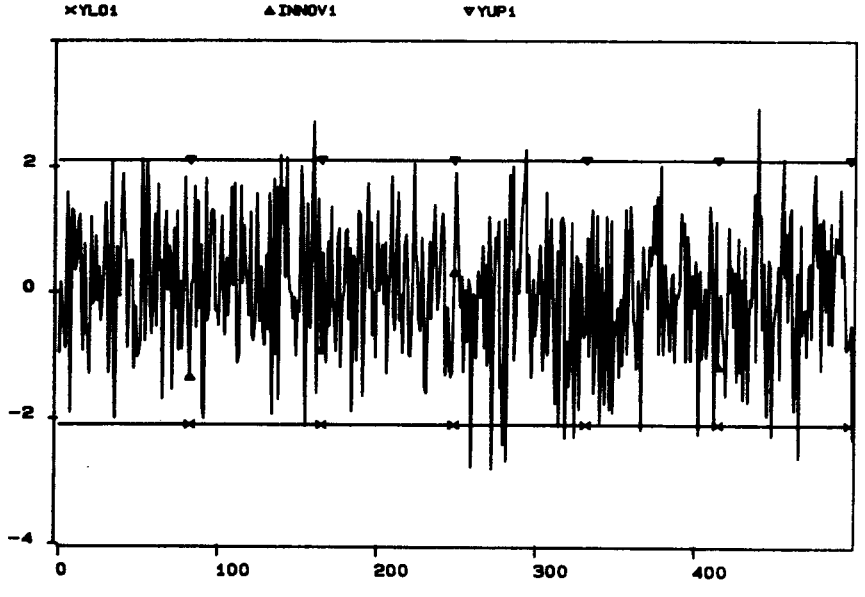

Fig. 7.9
The residuals and the 5 % limits as function of the time-instant k.

To check the innovation vector on being zero mean white noise, we use WTEST which gives the following results:

sample mean	bound mean	out of bound
0.048	0.093	8=3.21%

Table 7.7
The results from WTEST using the system with adapted noise matrices

The innovation vector is zero mean and white noise. The Kalman filter is optimal. The resulting values of the correlation function as function of the lag for the optimal filter are clearly better (fig. 7.10):

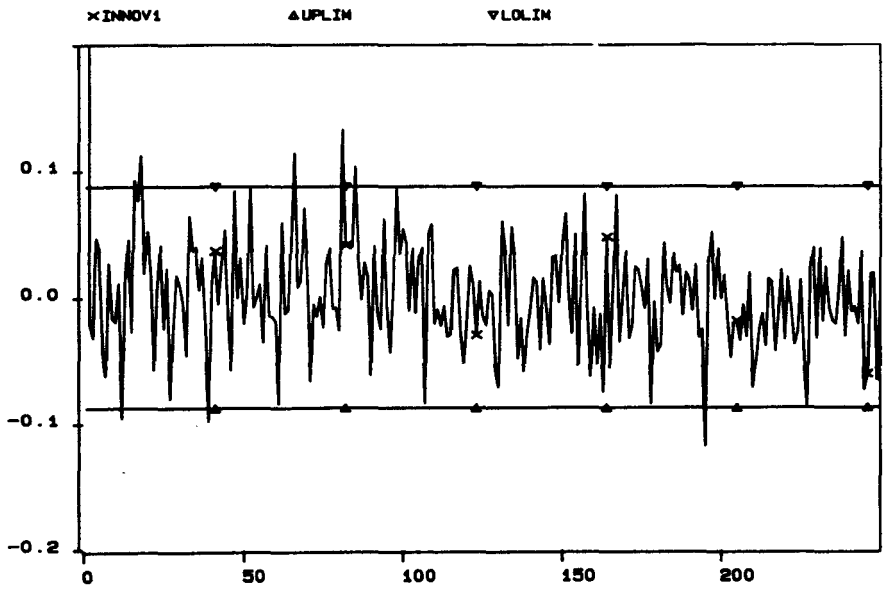

Fig. 7.10
The correlation functions versus lag for the optimal filter

This simple example shows the importance of adaptive filtering techniques for the noise covariance matrices. In this simple case, one iteration is enough to obtain an optimal filter. For MIMO systems the number of iterations can be larger and the calculations for obtaining a new estimate are more difficult, but the profit is not only a better estimate, but also better control when using an LQG algorithm. This shows that serious attention should be given to the development of adaptive filtering routines before elaborating on the control routines.

7.3. KALMAN FILTER PERFORMANCE

For a simple SISO-system, the steady-state Kalman gain matrix **K**, the steady-state covariance matrix **P** and the number of iterations necessary to reach steady state are calculated for different values of **Q**/**R**. Also the influence of a varying $\mathbf{P}_0$ is discussed. The steady-state solution is computed with a relative error smaller than 1.e-5.
We use the following system:

$$x_{k+1} = 0.76\, x_k + w_k$$
$$y_k = 2\, x_k + v_k$$

The results are ordered in tables 7.8a-b

Q	0.1	1.0	2.0	5.0	10.0
Iteration	12	6	6	5	5
P	0.155	1.19	2.13	5.14	10.1
K	0.146	0.311	0.340	0.362	0.371

Table 7.8a
Effects of variations in **Q**, (with **R** = 1).

R	0.1	1.0	2.0	5.0	10.0
Iteration	4	6	6	8	10
P	2.01	2.13	2.24	2.48	2.76
K	0.375	0.340	0.311	0.253	0.199

Table 7.8b
Effects of variations in **R**, (with **Q** = 2).

From tables 7.8a-b it is clear that increasing **Q** or decreasing **R** leads to a larger Kalman gain **K**. And that decreasing **Q** or increasing **R** leads to a smaller **K**. This means that when the system noise becomes larger, relative to the measurement noise, the difference between the calculated and realized measurement is weighted heavier. And that when the system noise becomes smaller, relative to the measurement noise, the model is trusted more.
Increasing **Q** and/or **R** leads to a larger covariance matrix **P** and decreasing **Q** and/or **R** leads to a smaller **P**. More uncertainty in the system leads to more uncertainty in the estimate.
The number of iterations necessary to reach steady-state with a relative error of 1.e-5 decreases when **Q** increases or when **R** decreases. The algorithm depends less on the model

and since the model has a slow impulse response this means that steady-state is reached faster.
When the ratio **Q/R** remains the same (**Q** = 2 in table 7.8a and **R** = 1 in table 7.8b), the Kalman gain matrix **K** remains the same. **Q/R** will be shown to be the relevant factor for the value of the Kalman gain matrix.

It can be shown that the steady-state filter gain **K**,

for a slightly more general system (1 input, 1 output, 1 state) :

$$x_{k+1} = \alpha\, x_k + w_k$$
$$y_k = 2\, x_k + v_k$$

can be written as:

$$\mathbf{K} = \frac{1}{\alpha}\{\sqrt{(\mathbf{Q/R})^2 + \frac{(\alpha^2-1)^2}{16} + \mathbf{Q/R} + \tfrac{1}{2}(\alpha^2-1)\mathbf{Q/R} + \frac{(\alpha^2-1)}{4}} - \mathbf{Q/R}\}$$

K evidently only depends on the ratio **Q/R**. It can be proven that also for MIMO systems, **K** is determined only by **Q/R**.

What is the influence of increasing the initial covariance matrix $\mathbf{P}_0$ on the staedy-state solutions? Since $\mathbf{P}_0$ enters in no formula for the steady-state solution for **P** and **K**, a different $\mathbf{P}_0$ will result in a different magnitude for the transient characteristic, but the steady-state conditions are unaffected. The length of the start-up period depends on the system matrix **A** and the initial covariance matrix $\mathbf{P}_0$.

This SISO system is only a very simple example. MIMO systems are more complicated. The SISO system is only used to show the influence of certain parameters on the behaviour of the steady state Kalman filter gain and the steady state covariance matrix. This behaviour explains a lot about how a Kalman filter basically behaves.

7.4. SUBOPTIMAL FILTERING

In this section, the developed modules will be used to test the optimality of the Kalman filter with variations in **Q** and **R**. Since we do not always know beforehand the exact values of the noise covariance matrices **Q** and **R**, it is important to know how to extract from the results presented to the user by KALSIM, KALEST and WTEST, the necessary information to determine if the Kalman filter is optimal. Optimality is easy to test, the innovations have to be zero mean white noise with a variance equal to the by Kalman predicted variance. It will be shown that it is very difficult to interpret the presented results when a Kalman filter functions suboptimally. Sometimes the Kalman filter is functioning as if it is an optimal filter, the innovations are zero mean, but the estimated and calculated variances do not agree. How should this be interpreted?
We have used the same system and the same *a priori* information to simulate 3 systems, one with large measurement noise, one with equal state and measurement noise and one with large state noise.
We have calculated the Kalman filter for the exact **Q**, and **R** and for 4 cases in which **Q** or **R** are varied, resulting in a suboptimal filter.

The used model is:

$$x_{k+1} = 0.97\, x_k + u_k + w_k \qquad 7.5a$$

$$y_k = 2\, x_k + v_k \qquad 7.5b$$

With the following *a priori* information:

$$P_0 = 1.e-12$$

$$x_0 = 2.5$$

The simulation uses 500 samples. This means that the autocorrelation function is calculated (by WTEST) for 250 lags. KALEST uses all 500 samples to calculate the bounds and the number of observations that are "out of bound". WTEST uses 250 lags of the autocorrelation function to calculate the number of observations that are "out of bound".

7.5. LARGE MEASUREMENT NOISE, SMALL STATE NOISE

In the first case the measurement noise is large compared to the state noise.

$$\mathbf{Q} = 1.e-4$$

$$\mathbf{R} = 4.$$

This is an extreme case of a system with very little state noise compared to the output noise.

7.5.1. KALMAN FILTER

The Kalman filter for the five different cases is calculated by KALMAN. The results are collected in table 7.9:

CASE	Q	R	P	K	W
1	1.e-4	4	1.64e-3	7.98e-4	4.01
2	1.e-4	40	1.69e-3	8.18e-5	40.0
3	1.e-4	0.5	1.43e-3	5.50e-3	0.51
4	1.e-1	4	3.37e-1	1.22e-1	5.35
5	1.e-8	4	1.69e-7	8.21e-8	4.00

Table 7.9
Results from the Kalman filter module KALMAN.
Large measurement noise, small state noise.

P is the steady-state covariance matrix, **K** the steady state Kalman gain matrix and **W** the steady-state covariance matrix of the innovation or residual sequence $\mathbf{r}_i$, all predicted by KALMAN.
As can be seen from table 7.9, case 1 uses the exact values for **Q** and **R**. In case 2-3 **R** is varied, in case 4-5 **Q**. In case 3-4 **Q/R** is larger than the optimal value in case 2-5 smaller. It can be observed that **W** is very sensitive to the value of **R** and relatively unsensitive to variations in **Q**. **P** is more sensitive to variations in **Q** than to variations in **R**. The Kalman gain matrix is very sensitive to variation in both **Q** and **R**.

7.5.2. STATE ESTIMATION

We use KALEST to estimate the state vector, using the by KALMAN calculated Kalman gain. The results for the innovation vector are collected in table 7.10

CASE	out of bound	estimated variance	calculated variance
1	4.6%	4.01	4.0
2	0.0%	**40.0**	4.0
3	**50.4%**	**0.51**	4.0
4	2.2%	**5.35**	**4.5**
5	4.4%	**4.00**	4.0

Table 7.10
The results for the innovation vector (obtained from KALEST).
Large measurement noise, small state noise

It can be observed that the calculated variance for all the four suboptimal cases is close to the one predicted by the Kalman filter for the optimal case. The estimated variance varies heavily with variations of **R** and only little with variations of **Q**. When **R** is too small, the model relies too much on the measurements which leads to 50.4 % out of bound. Compared to the optimal case, the suboptimal filters behave close to optimal.

7.5.3. WHITENESS TEST

The calculated innovation vector is tested on being a zero mean, white noise sequence by WTEST. The results are collected in table 7.11. The sample mean, which is the calculated mean of the innovation sequence, should be smaller than the bound mean, which is the maximum value for the absolute sample mean so that the sequence can statistically be called "zero mean". The number of elements of the correlation function that are "out of bound" should be smaller than 5 % for the sequence to be white. If a sequence complies to both of these conditions, it can be called a zero mean, white noise sequence. All the sequences are zero mean white noise.

CASE	sample mean	bound mean	out of bound
1	0.10	0.176	4.8%
2	0.11	0.176	4.8%
3	0.079	0.176	4.4%
4	0.006	0.187	4.8%
5	0.11	0.18	4.8%

Table 7.11
Results from the whiteness test module WTEST.
Large measurement noise, small state noise.

Even more information can be obtained by considering the first $n+1$ components of the correlation function, calculated by WTEST (stored in WTEST.DATA). These components are used by adaptive estimation algorithms to correct the noise covariance matrix **R** and **Q** (Table 7.12).

CASE	C_0	C_1	scale factor
1	1	0.313e-2	4.03
2	1	0.380e-2	4.03
3	1	-0.432e-2	40.4
4	1	**-0.130**	4.54
5	1	0.450e-2	4.03

Table 7.12
The first $n+1$ elements of the correlation function.
Large measurement noise, small state noise.

The 95 % limit for all cases is 8.877e-2 ($1.96/N^{½}$). When the state noise is too large (case 4), C_1 is larger than the allowed limit. Combining the results from table 7.9-12 lead to the following observations:

Case 1 (Optimal case) :

The number of "out of bound" is in both cases smaller than 5 percent. The calculated mean and the estimated mean are the same and the sample mean of the innovation vector is smaller than its bound mean. Case 1 is optimal (as it should be).

Case 2 (**R** too large) :

The estimated variance is larger than the observed variance due to the fact that **R** is too large. The innovation sequence is still zero mean, white noise. The calculated variance for the innovations is equal to the optimal case.

Case 3 (**R** too small) :

The percentage "out of bound" for the innovation vector is 50.4 percent, the Kalman filter is relying too much on the model. The estimated variance is also smaller than the observed one, but equal to the one for the optimal case. The innovation is zero mean, white noise.

Case 4 (**Q** too large) :

The estimated variance is larger than the observed variance and C_1 is "out of bound", the filter is relying too much on the measurements. The innovation sequence is zero mean, white noise. The calculated variance is still close to the estimated variance for the optimal case.

Case 5 (**Q** too small) :

The estimated and calculated variance are the same as for the optimal case. The innovations are zero mean white noise.

Conclusions:

The innovations are for all four suboptimal cases zero mean, white noise and since, except for case 4, the calculated variance for the innovations is equal to the, by KALMAN, estimated variance, they should be considered optimal. The noise covariance matrix **Q** for the optimal case and thus for the simulated data is so small that a large variation in **K** (table 7.9) is allowed without the filter performing far from optimally. But for each seperate case, the estimated and the calculated variance differ which means that the noise covariance matrices are not exact. In case of a small **Q**/**R** for the optimal case, variations in **Q** or **R** will cause the filter to function still close to optimal, despite the fact that there is an inconsistency between the estimated and calculated variance. But this is only a sign for incorrect noise matrices and does not imply that the resulting filter is performing worse.

7.6. EQUAL STATE AND MEASUREMENT NOISE

The state noise and the measurement noise are equal in this case:

$$\mathbf{Q} = 1.$$
$$\mathbf{R} = 1.$$

7.6.1. KALMAN FILTER

The Kalman filter is calculated by KALMAN, the results are in table 7.13:

CASE	Q	R	P	K	W
1	1.	1.	1.19	0.401	5.78
2	1.	10.	2.06	0.219	18.3
3	1.	0.001	1.00	0.484	4.01
4	10.	1.	10.2	0.473	41.9
5	0.001	1.	0.047	0.077	1.19

Table 7.13
Results from the Kalman filter module KALMAN.
Equal measurement and state noise.

As can be seen from table 7.13, case 1 uses the exact values for **Q** and **R**. In case 2-3 **R** is varied, in case 4-5 **Q**. For cases 3-4, **Q/R** is larger than for case 1, for cases 2-5 smaller. **P** seems mainly to be influenced by variations in **Q** and to a lesser extent by variations in **R**. Both **K** and **W** are effected by variations in **Q** and **R**.

7.6.2. STATE ESTIMATION

We use KALEST to estimate the state vector, using the by KALMAN calculated Kalman gain. The results for the innovation vector are presented in table 7.14:

CASE	out of bound	estimated variance	calculated variance
1	4.6%	5.78	5.7
2	0.0%	**18.3**	**6.6**
3	**10.4%**	**4.01**	**5.9**
4	0.0%	**41.9**	5.8
5	**52.0%**	**1.19**	**100.0**

Table 7.14
The results for the innovation vector (obtained from KALEST)
Equal measurement and state noise.

The estimated variance in case 3 and case 5 are smaller than the real variance, leading to more than 5 % out of bound. In case 2 and 4 the estimated variance is too large leading to 0.0 % out of bound. For cases 2-3-4, the calculated variance is still close to the estimated variance for the optimal filter.

7.6.3. WHITENESS TEST

The calculated innovation vector is tested on being a zero mean, white noise sequence by WTEST. If **Q/R** is smaller than for the optimal case (cases 2-5), the model is trusted to much and part of the state noise, filtered by the system, is added to the innovations as coloured noise. The innovations are not white anymore (table 7.15):

CASE	sample mean	bound mean	out of bound
1	0.087	0.21	2.4%
2	0.15	0.23	**6.4%**
3	0.072	0.21	4.0%
4	0.074	0.21	4.0%
5	**0.39**	0.29	**17.3%**

Table 7.15
The results form the whiteness test module WTEST.
Equal measurement and state noise.

Table 7.16 contains the first two values of autocorrelation function. For all four cases C_1 is out of bound. C_1 is used by adaptive filtering routines to correct **Q** and **R**.

CASE	C_0	C_1	factor
1	1	0.010	5.7
2	1	**0.374**	66.2
3	1	**-0.148**	5.9
4	1	**-0.129**	5.8
5	1	**0.663**	100.0

Table 7.16
The first n +1 elements of the correlation function.
Equal measurement and state noise.

The 95 % limit for all cases is 8.877e-2 ($1.96/N^{½}$). Combining the results from table 7.13-16 leads to the following observations:

Case 1 (optimal case) :

The number of "out of bound" is in both cases smaller than 5 percent. The calculated mean and the estimated mean are the same and the sample mean of the innovation vector is smaller than its bound mean. Case 1 is optimal.

Case 2 (**R** too large) :

The large **R** leads to too much reliance on the model. The innovation sequence is not white anymore as can be seen from table 7.15.

Case 3 (**R** too small) :

When **R** is too small, the estimated variance is also too small which leads to more than 5 percent out of bound. The resulting innovation sequence is still white and of zero mean.

Case 4 (**Q** too large) :

A too large **Q** leads to too much reliance on the measurements. The estimated variance is too large (0.0 percent out of bound) but the innovations are still white noise and zero mean.

Case 5 (**Q** too small) :

The filter relies too much on the model, the estimated variance is too small. The result is that more than 5 percent is out of bound and the innovations are not white and not of zero mean anymore.

Conclusions:

In case of equal state and measurement noise, using a smaller **Q/R** (case 2-5) results in a coloured noise component in the innovations. The model is trusted to much and part of the white state noise is filtered by the system and added to the innovations. Using a **Q/R** that is too large results in a filter with white innovations and with a calculated variance close to the estimated variance for the optimal filter. There is a statistical discrepancy between the calculated and estimated variance for each case, but compared with the optimal case, a larger **Q/R** performs still close to optimal. Overestimating the accuracy of the model or the inaccuracy of the measurements however, results in a filter that behaves badly (the innovations are not white and the calculated variance differ significantly from the estimated variance for the optimal filter).

7.7. LARGE STATE NOISE, SMALL MEASUREMENT NOISE

In this case the state noise is large compared to the measurement noise:

$$\mathbf{Q} = 4.$$
$$\mathbf{R} = 1.e-4$$

This is an extreme case of a system with measurements, very accurate compared to the state noise.

7.7.1. KALMAN FILTER

The Kalman filter is calculated by KALMAN, the results are in table 7.17:

CASE	Q	R	P	K	W
1	4.	1.e-4	4.00	0.485	16.0
2	4.	1.	4.22	0.458	17.9
3	4.	1.e-8	4.00	0.485	16.0
4	40.	1.e-4	40.0	0.485	160.0
5	0.1	1.e-4	0.10	0.485	0.4

Table 7.17
The results from the Kalman filter module KALMAN.
Small measurement noise, large state noise

As can be seen from table 7.17, case 1 uses the exact values for **Q** and **R**. In case 2-3 **R** is varied, in case 4-5 **Q**. For the cases 2-5, **Q/R** is smaller than the optimal value, for case 3-4 larger. The Kalman gain matrix **K** is unsensitive to variations in both **Q** and **R**. **W** and **P** are only sensitive to variations in **Q** .

7.7.2. STATE ESTIMATION

We use KALEST to estimate the state vector, using the by KALMAN calculated Kalman gain. The results are shown in table 7.18

CASE	out of bound	estimated variance	calculated variance
1	3.4%	16.0	15.0
2	3.2%	**17.9**	15.0
3	3.4%	16.0	15.0
4	0.0%	**160.0**	15.0
5	**74.0%**	**0.4**	15.0

Table 7.18
The results for the innovation vector (obtained from KALEST).
Small measurement noise, large state noise.

The calculated variance is the same for all 5 cases, due to the fact that **K** is the same. The estimated variance is mainly influenced by variations in **Q**, **R** is too small to have a real influence.

7.7.3. WHITENESS TEST

The calculated innovation vector is tested on being a zero mean, white noise sequence by WTEST. The results are in table 7.19:

CASE	sample mean	bound mean	out of bound
1	0.14	0.34	2.4%
2	0.15	0.34	3.2%
3	0.14	0.34	2.4%
4	0.14	0.34	2.4%
5	0.14	0.34	2.4%

Table 7.19
The results from the whiteness test module WTEST
Small measurement noise, large state noise.

In all cases the innovations are white noise and zero mean.

Table 7.20 contains the first two values of the calculated autocorrelation function, C_0 and C_1 :

CASE	C_0	C_1	factor
1	1	0.054	15.0
2	1	**0.106**	15.0
3	1	0.054	15.0
4	1	0.054	15.0
5	1	0.054	15.0

Table 7.20
The first n +1 elements of the correlation function.
Small measurement noise, large state noise.

The 95 % limit for all cases is 8.877e-2 ($1.96/N^{1/2}$). A too large **R** results in a too large C_1. Combining the results from table 7.17-20 leads to the following observations:

Case 1 (Optimal case) :

As it should be, case 1 is optimal. The estimated and calculated variances are equal, the innovations are white noise zero mean and less than 5 % out of bound.

Case 2 (**R** too large) :

Case 2 seems to be close to being optimal. The innovations are white and of zero mean, the estimated and calculated variance are almost the same, only C_1 is an indication that the model is trusted too much.

Case 3 (**R** too small) :

Case 3 is optimal, not in the sense that it has the right **R** but in the sense that the innovations are white and zero mean and the variances are the same. It uses the same Kalman gain matrix as the optimal case.

Case 4 (**Q** too large) :

Case 4 is optimal except for a too large estimate for the variance. The Kalman gain matrix is the same despite the fact that **Q** is too large.

Case 5 (**Q** too small) :

Also case 5 is optimal except for a too small estimate for the variance, leading to more than 5 % out of bound.

Conclusions:

It is obvious that all 5 cases perform the same, since **K** is almost the same. The sensitivity of **K** for changes in the noise matrices is very small. There is only a statistical inconsistency between the calculated and estimated variances but the filter performs the same.

7.8. CONCLUSIONS

Three different cases have been considered: **Q/R** << 1, **Q/R** = 1, and **Q/R** >> 1. When **Q/R** >> 1, the performance of the Kalman filter is very insensitive to incorrect noise covariance matrices, the Kalman gain matrix **K** does not vary a lot and the innovations are zero mean white noise. The estimated and calculated variances are different but the filter is still close to optimal.
When **Q/R** << 1, the innovations are also zero mean white noise when incorrect noise matrices are used. Only when **Q/R** = 1, the Kalman filter is very sensitive to incorrect noise matrices especially when the model is trusted more than it should. In this case, the resulting innovations show a coloured noise component. The state noise, after being filtered by the system is coloured noise and is as such added to the innovations. **Q** and **R** need to be determined accurately to prevent this. Generallt can be stated that it is very difficult to determine from the by KALMAN, KALEST and WTEST presented results, how to adapt the noise covariance matrices. Adaptive filtering routines can perform this task much more efficiently The innovation correlation method developed by Carew seems to be the most promising one. It is proven to be numerically more stable than the method developed by

Mehra, it uses innovation correlation functions which have already been calculated by WTEST and it is proven to converge assymptotically to the exact values for **Q**, **R** and **K**.

Consequently, in the case of unknown noise covariance matrices:

(1) Use some sensible estimates for **Q** and **R** and calculate the corresponding Kalman filter using KALMAN.

(2) Use KALEST to obtain the innovation sequence, the 95 % limits, the number of observations that are "out of bound" and the observed variances. A comparison of the observed variances with the by KALMAN estimated ones, shows if the estimates for **Q** and/or **R** are too small or too large. But only an adaptive filtering routine can determine how much **Q** and **R** should be changed.

(3) Use WTEST to test the innovations on being zero mean white noise. A coloured noise component means that the model is trusted too much (**Q** too small or **R** too large).

(4) The Kalman filter is performing optimally when the innovations are zero mean white noise, when the by KALEST observed variances are equal to the by KALMAN predicted ones, and when less than 5 % of the observations of the innovations are "out of bound".

(5) If the innovations are not zero mean white noise or the estimated and observed variances do not agree, use an adaptive filtering routine to obtain better estimates for **Q**, **R** and **K**.

Chapter 8

CONCLUSIONS AND SUGGESTIONS

8.1. CONCLUSIONS

- A set of PRIMAL modules has been developed to design a Kalman filter for a stochastic state space model and to test and evaluate its performance.

- The performance of the Kalman filter can be evaluated without knowing the "true states". WTEST tests if the innovations are zero mean white noise. KALEST calculates the variance for the observed innovations and the number of innovations that are "out of bound", and KALMAN calculates an estimate for the variance of the innovations. The Kalman filter is functioning optimally when:

(1) the innovations are zero mean white noise which means that the calculated mean is smaller than the bound mean (zero mean test) the number of values of the autocorrelation function that are "out of bound" is smaller than 5 percent (whiteness test)

(2) the estimated and calculated variance are equal.

If not, it is still possible that the filter is close to optimal depending on the actual values of **Q**, and **R**.

- Apart from modules for designing and evaluating the Kalman filter, a module for solving the Riccati equation has been developed. Together these modules form the first controller design facility in PRIMAL.

- In the case of unknown noise covariance matrices the adaptive filtering techniques described in chapter 5 may be used. Chapter 7.3-5 show that it is difficult to determine **Q** and **R** from KALSIM, KALMAN, KALEST and WTEST even for a SISO system. The innovation correlation method developed by Carew (eqn. 5.35-37) [CAR73] is the most promising, it requires no solution of a Lyapunov type of equation and it seems to perform better than the method developed by Mehra (eqn. 5.27-32) [MEH70], [MEH72] and is numerically more attractive.
The output correlation method does not require any initial estimates for **Q** and **R**, but it requires a stable system matrix **A** or $\mathbf{y}_k$ to be stationary. Furthermore the correlations for the innovations have already been calculated in the whiteness test and are therefore available to the user.
The covariance matching methods seem simple and straightforward, but convergence of these methods has never been proven and they only seem to work well if the state covariance matrix **Q** is known and **R** has to be estimated.

- The matrices necessary for the optimal control **E** and **F** are difficult to determine. The

user has to formulate his objectives in terms of desired trajectories for the state vector and allowed values for the input vector. The algorithm does not guarantee these objectives, but by repeating this step with different matrices it should be possible to design a feedback matrix which meets the specifications. Minimization of the output vector $\mathbf{y}_k$ can be achieved by setting:

$$\mathbf{E} = \mathbf{C}\mathbf{C}^T$$
$$\mathbf{F} = \mathbf{D}\mathbf{D}^T$$
$$\mathbf{M} = \mathbf{C}\mathbf{D}^T$$

This means that the user does not have to specifiy the matrices, they are inherent to the used system. How this method performs can be evaluated after implementation of the simulator.

8.2. SUGGESTIONS

- Before the LQG-module will be used in practice it is necessary to implement an adaptive filtering routine and a simulator for evaluating the performance of the LQG-feedback matrix using a Kalman filter for state estimation. Since optimal performance of the LQG-algorithm is assured only when all the previous steps are optimal, it is important to develop an adaptive filtering routine first.

- The possible use of the residuals or innovations are numerous. Apart from being able to use them for adapting the noise covariance matrices, they can also be used for sensor failure detection or spike filtering (on line). This can prevent the control routine from using spikes. Special routines may warn the user of a possible sensor failure or change in the system model. The user can take the necessary actions to prevent further damage.

- Mehra suggests that the estimate for the Kalman gain matrix **K** can be improved by repeatedly filtering the data with the adapted Kalman filter and using the new autocorrelation functions to obtain a better estimate. However, this is not always true. If we start with the optimal Kalman gain and there are any errors in the estimation of the correlation functions, due to the finite sample size, the adaptive filtering routine will not lead to the optimal gain again. The value of Mehra's suggestion must be determined when an adaptive filtering routine is available.

- PRIMAL offers two possibilities for implementing the obtained controller. The first possibility is to implement in PRIMAL control routines that use the computer to control a process in real-time. The second possibility is to develop modules that approximate an optimal control regulator using conventional PID-controllers. The first possibility requires enough computing capabilities for the real-time controller to perform in real-time, but the obtained LQG-controller will be implemented exactly. The second possibillity will implement an approximation which means that the actual performance is difficult to predict. But the conventional PID-controllers are easier to implement and no computer time is required. The introduction of process computers in (industrial) processes will make the first option more attractive, but many processes are still controlled by conventional controllers.

REFERENCES

[AND71]
B.D.O. Anderson, J.B. Moore, *Linear optimal control*. New York: Prentice Hall, 1971.

[AND42]
R.L. Anderson, "Distribution of the serial correlation coefficient," *Ann. Math. Statist.*, vol. 13, 1942.

[BRA75]
K. Brammer, *Kalman-Bucy-filter, deterministische beobachtung und stochastische filterung*," München: R. Oldenbourg Verlag, 1975.

[CAR73]
B. Carew, P.R. Belanger, "Identification of optimum filter steady-state gain for systems with unknown noise covariances," *IEEE Trans. of Autom. Contr.*, vol. AC-18, pp.582-587, Dec. 1973.

[DAM86]
A.A.H. Damen, A.J.W. van den Boom, *Toegepaste systemanalyse*, syllabus Dept. of Elec. Eng., Techn. Univ. Eindhoven, Netherlands, 1986.

[DOO81]
P. van Dooren, "A generalized eigenvalue approach for solving Riccati equations," *SIAM J. Sci. Stat. Comput.*, 2, pp. 120-135, 1981.

[FRI86]
B. Friedland, *Control system design. an introduction to state-space methods* , New York: McGraw Hill, 1986.

[GER87]
P.H.M. Gerlings, *The Hankel realization method in PRIMAL*, M.Sc. Thesis (NR-1485P) System and control group dept. Techn. Fys., Techn. Univ. Eindhoven, 1987.

[GOD74]
S.S. Godbole, "Kalman filtering with no *a priori* information about noise: white noise case: identification of covariances, " *IEEE Trans. Autom. Contr.* (short paper), vol. AC-19, pp. 561, Oct. 1974.

[GOO84]
G.C. Goodwin, K.S. Sin, *Adaptive filtering, prediction and control*, New Yersey: Prentice Hall, 1984.

[HAN60]
E.J. Hannan, *Time series analysis*, New York: Wiley, 1960.

[HAU79]
M.L.J. Hautus, *Lineaire multivariabele systemen*, Syllabus Math. Dept., Techn. Univ. Eindhoven, Netherlands, 1987.

[KAI68]
T. Kailath,"An innovations approach to least-squares estimation, part I:linear filtering in additive white noise," *IEEE Trans. Autom. Contr.*, vol. AC- 13, pp. 646-655, Dec. 1986.

[KAL60]
R.E. Kalman, "A new approach to linear filtering and prediction problems," *Trans ASME*, J. Basic Eng., Series 82D, pp. 35-45, 1960

[KAL61]
R.E. Kalman, R.S. Bucy, "New results in linear filtering and prediction theory," *Trans. ASME, J. Basic Eng.*, Series 83D, pp.95-108, Mar. 1960.

[KWA72]
H. Kwakernaak, *Linear optimal control systems*, New York: John Wiley, 1972.

[LEE64]
R.C.K. Lee, Y.C. Ho, "A Bayesian approach to problems in stochastic estimation and control," *IEEE Trans. Autom. Contr.*, vol. AC-9, pp.333-339, Oct. 1964.

[LIN87]
R. v.d. Linden, *A Markov impulse response in PRIMAL.*, internal memo (NR1284P) System and Control group, dept. of Techn. Fys., Technical Univ. Eindhoven, 1987.

[MAR74]
W.C. Martin, A.R. Stubbero, "An additional requirement for innovations," *IEEE Trans. Autom. Contr.*, vol. AC-19, pp. 583-584, Oct. 1974

[MAY79]
P.S. Maybeck, *Stochastic models, estimation, and control, vol.* I, Vol. 141 Mathematics in Science and Engineering, New York, Academic Press, 1979.

[MEH70]
R.K. Mehra, "On the identification of variances and adaptive Kalman filtering," *IEEE Trans. Autom. Contr.*, vol. AC-15, pp.175-184, Apr. 1970.

[MEH72]
R.K. Mehra, "Approaches to adaptive filtering," *IEEE Trans. Autom. Contr.* (short papers), vol. AC-17, pp. 693-698, Oct. 1972.

[MEU88a]
P. van Meurs, *KALSIM, a stochastic state space simulator in PRIMAL*, Internal memo (NR-1504P) System and Control Group, Dept. of Techn. Fys., Technical Univ. Eindhoven, 1988.

[MEU88b]
P. van Meurs, *KALMAN, a Kalman filter module in PRIMAL*, Internal memo (NR-1505P) System and Control Group, Dept. of Techn. Fys., Technical Univ. Eindhoven, 1988.

[MEU88c]
P. van Meurs, *KALEST, a state estimation module in PRIMAL.*, Internal memo (NR-1506P) System and Control Group, Dept. of Techn. Fys., Technical Univ. Eindhoven, 1988.

[MEU88d]
P. van Meurs, *WTEST, a whiteness test and zero mean test module in PRIMAL*, Internal memo (NR-1507P) System and Control Group, Dept. of Techn. Fys., Technical Univ. Eindhoven, 1988.

[MEU88e]
P. van Meurs, *RIC, a Riccati equation solver in PRIMAL*, Internal memo (NR-1508P) System and Control Group, Dept. of Techn. Fys., Technical Univ. Eindhoven, 1988.

[MOL73]
C. Moler, G. Stewart, "An algorithm for generalized matrix eigenvalue problems," *SIAM J. Numer. Anal.*, 10, pp. 241-256, 1973.

[MYE76]
K.A. Myers, B.D. Tapley, "Adaptive sequential estimation with unknwon noise statistics," *IEEE Trans. of Autom. Contr.*, vol AC-21, pp. 520-523, 1976.

[OLC83]
S. Olcer, *Commande sous optimale adaptive de systèmes stochastiques linéaires*, Thesis Dept. Mech. Eng., Institution d'Automatique de L'EPFL, Lausanne, France, 1983.

[PRI68]
C.F. Price, J.J. Reyst, "Conditions for assymptotical stability of the discrete minimum variance linear estimator," *IEEE Trans Autom. Contr.* (short papers), vol AC-13, pp.702-705, Dec. 1968

[SCH77]
K.W. Schrick, *Anwendungen der Kalman filter technik, anleitung und beispiele*, München: R. Oldenbourg Verlag, 1977.

[SOR85]
H.W. Sorenson, *Kalman filtering: Theory and application*, New York, IEEE press, 1985.