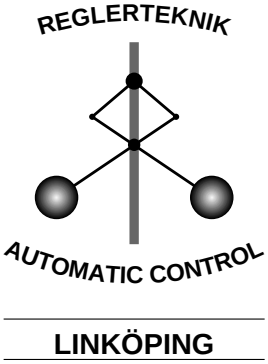
Linköping Studies in Science and Technology
Thesis No. 793

Aspects of the Identification of Wiener Models

Anna Hagenblad



Division of Automatic Control
Department of Electrical Engineering
Linköpings universitet, SE–581 83 Linköping, Sweden
WWW: `http://www.control.isy.liu.se`
Email: `annah@isy.liu.se`

Linköping 1999

Aspects of the Identification of Wiener Models

Department of Electrical Engineering,
Linköpings universitet,
SE–581 83 Linköping,
Sweden.

ISBN 91-7219-629-7
ISSN 0280-7971
LiU-TEK-LIC-1999:51

Printed by UniTryck, Linköping, Sweden 1999

To my husband

Abstract

System identification deals with the problem of constructing models of systems from observations of the inputs and outputs to the systems. In this thesis, a particular class of models, Wiener models, is studied. The Wiener model consists of a linear dynamic block, followed by a static nonlinearity.

The prediction error method is formulated for the Wiener model case, and it is discussed how the predictor depends on the noise assumptions. It is shown that under certain conditions, the prediction error estimate is consistent. Conditions that certify consistency for a simplified, approximative predictor are also stated.

Consistent in theory, the prediction error estimate is much too complicated to calculate analytically in practice, and numerical methods must be used. Furthermore, the prediction error criterion may have several local minima, so a good initial estimate is needed. A considerable part of this thesis deals with how to calculate such an initial estimate.

By a particular choice of parameterization of the linear subsystem and the inverse of the nonlinearity, it is possible to formulate an error criterion where the parameters enter quadratically. It is discussed how this error criterion may be minimized using linear regression, quadratic programming or the total least squares method. This initial estimate may then be used in the numerical minimization of the prediction error criterion.

An algorithm for identification of Wiener models is presented, and it is shown that the algorithm under some conditions gives a consistent estimate. The algorithm is also applied to both simulated and experimental data.

Acknowledgments

The work presented in this thesis was supported by the Swedish Research Council for Engineering Sciences (TFR), which is gratefully acknowledged, but writing a thesis (even a licentiate one) takes more support than only financial.

First, you need a supervisor. It is of course important that your supervisor has some knowledge of the area you are working in, and it is even better if he or she is an expert in the field. Visions for future research will help you choose a good subject, and international contacts will provide you with perspectives. However, I doubt that this would be enough, if he or she did not have the ability to guide you in your research by insightful comments and suggestions, encouragement when you feel stuck, and appreciation of things you had forgotten that you did well.

My supervisor, Professor Lennart Ljung, has all of the above. I am grateful to him for drafting me to the Automatic Control group, and for guiding me this far in my research. He has also proof-read my manuscript and has come with suggestions for improvement more times than I thought was possible for someone with a limited time and/or patience.

A supervisor is not everything. Without the discussion with colleagues and their help in proof-reading your manuscript, the quality of the final product would be significantly lower. Dr. Niclas Bergman, Dr. Peter Lindskog and Fredrik Tjärnström has read the manuscript and given their suggestions on how to improve it. Måns Östring has had valuable opinions on the design.

Other aids when working on a thesis: The ECSEL[1] graduate school, with its abundant supply of graduate courses. Dr. Niclas Bergman and Dr. Anders Stenman, who maintained the LaTeX and XEmacs installations, respectively. Mattias Olofsson, who kept the computers running. Coffee break discussions, floorball games and junk seminars, all contributing to the atmosphere.

Finally, I want to mention my family, whose support is crucial to me. My mother, who thoroughly researches my English questions. My father, a source of knowledge whenever it comes to literature. My sister Jenny, dear to play, fight and argue with, who asked the important question "What is a system?"[2]. And Jan-Erik, who always encourages me with his love, support and never-ending faith in me. He has also read this whole thesis. I am proud of you, and I love you.

[1] Excellence Center for Computer Science and Systems Engineering in Linköping

[2] see page 1

Contents

Notation

Symbols

Symbol	Explanation	See page
u, $u(t)$	The input signal to the linear system	9
x, $x(t)$	The input to the nonlinear system	9
y, $y(t)$	The output of the nonlinear system	9
e, $e(t)$	Measurement noise	9
v, $v(t)$	Process noise	9
$\hat{y}(t, \theta, \eta)$	The prediction of y given old inputs and outputs	10
Z^t	The set of input and output data, u and y, up to time t	10
X, Y	Stochastic vectors containing $x(1), \ldots, x(N)$ and $y(1), \ldots, y(N)$, respectively	23, 17
G, $G(q)$, $G(q, \theta)$	The linear dynamic subsystem	10
f, $f(\cdot)$, $f(\cdot, \eta)$	The static nonlinear subsystem	10
θ	The parameters of the linear system	9
η	The parameters of the nonlinear system	10
$\hat{\theta}$, $\hat{\eta}$	Estimates of the parameters	10

Symbol	Explanation	See page
θ_0, η_0	True values of the parameters	11
$\Theta = (\theta, \eta)$	The parameters of the Wiener system	23
q	The shift operator	9
E	Expectation operator	10
$\overline{E}$	$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} E$	12
$V_N(\theta, \eta)$	The prediction error criterion	10
$f_Y(\theta, \eta, \mu)$	The probability density function of Y	17
$p(Y\|\Theta)$	The likelihood of Y given Θ	24
$\varphi(t)$	Regressor vector	22
$\zeta(t)$	Instrumental variables	23
σ^2	Variance	25
A, B, C	State space matrices	13, 34
$B_i(y)$	B-splines basis functions	37
σ_i	Singular values	56
$\mathbf{R}$	The set of real numbers	65

Abbreviations

Abbreviation	Explanation	See page
AIC	Akaike's Information Criterion	60
ARX	Auto-Regressive with eXogeneous Input	32
BJ	Box-Jenkins model	33
EKF	Extended Kalman Filter	13
EM	Expectation Maximization	23
FIR	Finite Impulse Response	33
FPE	Akaike's Final Prediction Error	60
FSF	Frequency Sampling Filter	35
HH	Hinging Hyperplanes	39
IV	Instrumental Variables	22, 68
LR	Linear Regression	21, 67
NN	Neural Networks	38
OE	Output Error	33
PDF	Probability Density Function	17
PRBS	Pseudo Random Binary Signal	80
QP	Quadratic Programming	45
TLS	Total Least Squares	45, 69

1

Introduction

1.1 System Identification

This thesis deals with *system identification* , and a natural first question is then: What is a system? Generally speaking, a system is an object we want to study. It could be the Swedish economy, the learning rate of English students, or the calf mortality in highland cattle (Hagenblad, 1998b). Central in our perception of a system is the concept of inputs and outputs. An output is something of interest that we observe from the system. In the above examples, it could be the interest rate, the students' score on a test, and the percentage of calves dead four days after the birth. Things that affect the way the system behaves are called inputs or disturbances. Inputs are the ones we can control. In the English student example this could be the number of hours the students have studied, for the Highland cattle it could be the amount of vitamins and minerals fed to the animals. Disturbances also affect the system but we cannot control them. The study hours can be more or less efficient, but we can only measure how many they are, so the variations in efficiency are considered a disturbance. Also how much vitamins and minerals the Highland cattle eat will not be the same as how much they are fed; the variation can be thought of as a disturbance. We can have several disturbances, as well as several inputs and outputs of a system. Inputs, outputs, and disturbances are also called signals.

In system identification we want to find a mathematical relation between

the inputs, outputs and disturbances. This relation is what we call a *model*. We measure the inputs and the outputs, possibly also the disturbances, and we want to estimate, or identify, a model from our measurements. To be able to do this, we start with a *model structure*. A model structure is an idea about the relation between the input and the output. In the English student example, we might suspect that the more hours a student studies, the better he will score on the test. One possible model structure is then

$$y = ku \tag{1.1}$$

where y is the output, the score on the test, and u is the input, the number of hours studied. The constant k is an unknown parameter of the system.

To determine, or estimate, k, we collect measurements of y and u. Now suppose one student studied for 3 hours and scored 30 points on the English test, another one studied for 8 hours and scored 80 points. The estimate $k = 10$ is then consistent with our measured data. Of course this is a very simplified model. For one thing, a student's score on the test will also be affected by his prior knowledge. We may consider this as a disturbance and disregard it, or we may try to measure the prior knowledge and use that too in the model.

This is the essence of system identification: We are interested in a particular system. We decide what are the interesting outputs and inputs, and what disturbances we want to include in the model. We select a model structure. From measurements of the inputs and outputs, and possibly also disturbances, we estimate the parameters in our model structure. We validate our model, to see how accurate and useful it is, and we might then go back to select other inputs, outputs and disturbances, or another model structure, to estimate a better model.

1.2 The Wiener Model

This thesis treats models of a certain class: Wiener models. A Wiener model is depicted in Figure 1.1. It consists of a linear dynamic system G followed by a static nonlinearity f. The input u and the output y are measurable, possibly with noise, but we cannot measure the intermediate signal x.

To relate the Wiener model to our previous example with the English students, we suppose that we are interested in not only the result on a particular test, but on several tests, at different times, and we suspect that these results are related. We give a test every week, and we let $y(t)$ be the score on the test week t. Instead of relating this directly to the number of

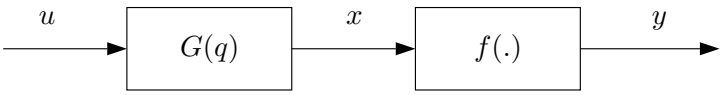


Figure 1.1: A Wiener model

hours studied that week, we will say that the score on a test is a function of the knowledge that week. We assume that there is such a thing as "knowledge", but we cannot measure it directly. The knowledge week t is denoted $x(t)$, and the test score $y(t) = f(x(t))$. This is the static nonlinear part of the Wiener model.

We now relate the knowledge x to the number of hours studied, u. The knowledge week t is of course a function of the number of hours studied week t, which we denote by $u(t)$, but also a function of the knowledge last week, week $t-1$, and the number of hours studied last week, $u(t-1)$. The following model structure might then be interesting:

$$x(t) = b_0 u(t) + b_1 u(t-1) + a_1 x(t-1) \tag{1.2}$$

This is the linear dynamic part of the Wiener model; the knowledge depends not only on the number of hours studied that week, but also other weeks. The constants b_0, b_1 and a_1 are the unknown parameters we want to estimate. The subject of this thesis is how to use measurements of u and y (number of study hours and test score) to estimate these parameters.

Wiener models naturally arise also in other situations. A linear system where the measurement device has a nonlinear characteristics is one example. In chemistry, pH control systems can be described as Wiener models (see Kalafatis et al., 1995; Pajunen, 1992). Hunter and Korenberg (1986) cites several biological examples. Zhu (1999a) uses a Wiener model to identify a distillation column. Boyd and Chua (1985) showed that a very large class of systems, time invariant systems with *fading memory*, can be approximated arbitrarily well with a Wiener model where x as well as u and y may be vectors. In this thesis we treat only scalar Wiener systems, i.e., x, u and y are scalars.

Our goal is to find a linear dynamic model relating u and x, and a nonlinear static one relating x and y. We will consider *parametric* models, where the output can be described as a function of the input and some parameters. Different models are described by different values of these parameters. We will also restrict ourselves to discrete time. For the linear dynamic system

from u to x, we will write this as:

$$x(t) = G(q, \theta)u(t) \tag{1.3}$$

where q is the time-shift operator, $qu(t) = u(t+1)$ and θ is a parameter vector describing the linear system. The nonlinear system relating $x(t)$ with $y(t)$ is described as

$$y(t) = f(x(t), \eta) \tag{1.4}$$

where f is a nonlinear function of $x(t)$, determined by the parameters η. We want to use measurements of the input u and the output y to estimate the parameters, both θ and η.

In estimating the parameters of the linear and nonlinear system we want to arrive at the "best" model in some sense. What is best depends of course on what we want to use the model for. One often used approach is the prediction error approach: we use the estimated model to predict the output for a given input. For given values of the parameters θ and η and a given input u we can calculate the predicted output, $\hat{y}$. $\hat{y}$ will depend on the parameters, as well as the time t, so we will denote it $\hat{y}(t, \theta, \eta)$.

To measure the quality of the estimate we will compare the predicted output with the measured one. We do this by forming the following prediction error criterion:

$$V_N(\theta, \eta) = \frac{1}{N} \sum_{t=1}^{N} (y(t) - \hat{y}(t, \theta, \eta))^2 \tag{1.5}$$

where $y(t)$ is the measured output at the time instance t, $\hat{y}(t, \theta, \eta)$ is the predicted output at the same time, and N is the number of data. We will say that the best estimate of θ and η is the one that minimizes $V_N(\theta, \eta)$, and we want to find these values of θ and η.

For a well-defined model structure and given measurements, $V_N(\theta, \eta)$ may be formed explicitely as a function of θ and η. It is normally too complicated to be minimized analytically, but there are reliable numerical methods available. The numerical methods use an initial estimate, a guess, of the parameter values to find a better estimate. A large part of this thesis deals with how to make a good initial estimate, and convert it to a form useful in the numerical minimization.

1.3 A Motivating Example

Wiener models are quite similar to linear models - just remove the nonlinear block. One might therefore ask how linear model identification perform on data from Wiener models. Suppose we do not know that our system is nonlinear, so we try to identify a linear model. A large number of well-known methods exist; we may try a prediction error method (Ljung, 1999) or a subspace-based method (van Overschee and De Moor, 1996). It can be shown that if the input is Gaussian, the linear estimate will be consistent. This relies on Bussgang's theorem (Bussgang, 1952): the cross-correlation of two Gaussian signals is proportional to the cross-correlation when one of them has undergone a nonlinear transformation. This can be applied to Wiener model: If the input $u(t)$ is Gaussian, so is the intermediate signal $x(t)$, and according to Bussgang's theorem, the cross-correlation between $u(t)$ and the output $y(t)$ will then be proportional to the cross-correlation between $u(t)$ and $x(t)$.

Modeling the system as linear also provides us with an approximation of the linear subsystem. This approximation can be used to simulate the intermediate signal x. If the linear model is reasonably accurate, we may plot the simulated $x(t)$ versus the measured $y(t)$, and get a visual representation of the nonlinearity. We can then estimate the nonlinearity from this plot, i.e., form the simulated and measured $\{x(t), y(t)\}$ data.

A small example will show that this is not necessarily the best procedure, and motivate the method that is the focus of this thesis. The example consists of a second order linear system, and an exponential nonlinearity. No measurement noise is added to the signals, to show the problems that can arise even in the noise free case.[1]

The system is given by the following equations:

$$x(t) = \frac{q^{-1}}{(1-\alpha q^{-1})^2} u(t) \quad \text{where } \alpha = 0.7 \tag{1.6}$$

$$y(t) = e^{x(t)} \tag{1.7}$$

The input signal is a sum of sinusoids:

$$u(t) = \sum_{k=1}^{20} \sin(k\pi t/10 + \phi_k) \tag{1.8}$$

[1] MATLAB code for all examples in the thesis can be downloaded from:
`http://www.control.isy.liu.se/publications/doc?id=1063`

where ϕ_k is a stochastic variable with uniform distribution on $[0, 2\pi]$.

First, the mean was removed from the output data set $\{y(t)\}$. An output error model $\widehat{G}$ was estimated from the set of input-output data, $\{u(t), y(t)\}$. The model $\widehat{G}$ was used to simulate the linear system. We thus got an estimate of the intermediate signal $x(t)$. To the left in Figure 1.2, the estimated $x(t)$ is plotted versus the measured $y(t)$. The true nonlinearity is also shown. The $(x(t), y(t))$ points are scattered around the true nonlinearity, but even though there is no noise, there is a significant error.

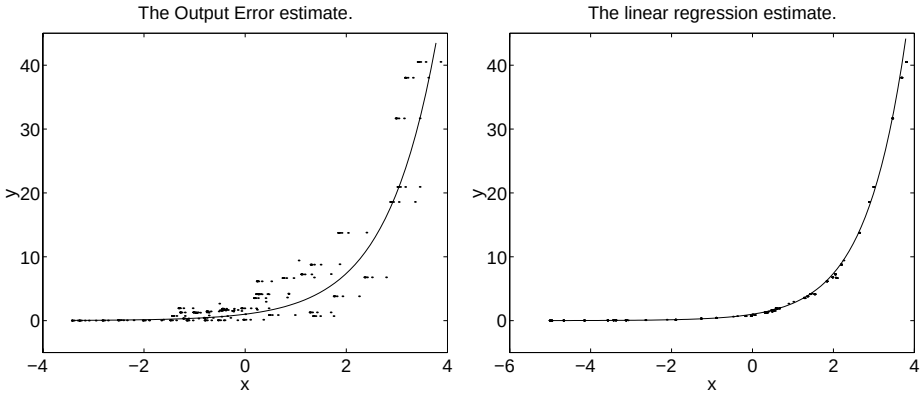


Figure 1.2: Estimates of the linear model are used to simulate the intermediate signal $x(t)$ and plot it against the measured output $y(t)$. To the left an output error model estimate is used; to the right a linear regression estimate, as described in Chapter 4. (The x-axis scales are slightly different in the two plots.) The solid line is the true nonlinearity.

The method discussed in this thesis shows that we can do better than just using a linear output error model. Let the linear system be parameterized by an FIR model, and the inverse of the nonlinearity with linear B-splines. We can then formulate an error criterion where the parameters enter linearly, and minimize the criterion with linear regression. The estimate obtained this way was used to simulate $x(t)$ and plot it against $y(t)$ to the right in Figure 1.2. The method is described in detail in Chapter 4. The data points are here much closer to the true nonlinearity.

1.4 Outline and Contributions

The outline of this thesis is as follows: The estimation problem is formulated as a prediction error minimization in Chapter 2, and the consistency of this approach is discussed. Some approaches to Wiener model estimation from the literature are also described. Different parameterizations of the two subsystems of the Wiener model are treated in Chapter 3. The numerical search methods are strongly dependent on a good initial estimate. How to obtain this is discussed in Chapter 4. The initial estimate is expressed as a certain parameterization, while another parameterization may be wanted in the final estimate. This change of parameterization could be seen as a model reduction. How to convert the initial estimate to the desired structure is treated in Chapter 5. Chapter 6 presents an algorithm to estimate a Wiener model, and analyzes the consistency of the estimates obtained from the algorithm. The algorithm is also depicted as a flowchart in Figure 1.3, with references to the different parts of the thesis. In Chapter 7, the algorithm is applied to several examples, with both real and simulated data.

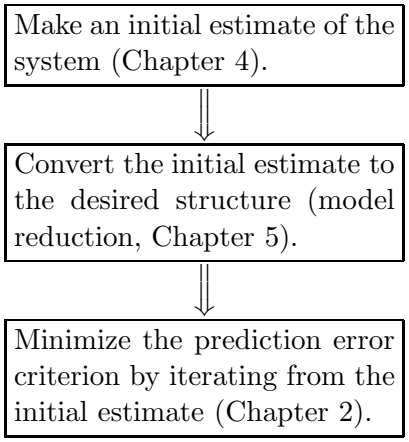


Figure 1.3: Flowchart for Wiener model estimation as presented in this thesis.

The main contributions of the thesis can be summarized as follows:

- The Wiener Model Identification Algorithm in Chapter 6, and the consistency analysis thereof. The initial linear regression estimate derived in Chapter 4 is similar to the estimate presented in Kalafatis et al. (1997). The prediction error minimization method is well known

from various sources, see e.g. Ljung (1999) for linear systems, Sjöberg et al. (1995) for nonlinear systems.

- The overview of how the nonlinearity affects the predictor, and the consistency for some simplified predictors, Section 2.2.
- The model reduction needed to go from the initial estimate to the prediction error criterion, presented in Chapter 5. The linear part is however well known.

Parts of the results in this thesis have been presented at different conferences:

Hagenblad, A. (1998a). Identifiering av Wienermodeller. In *Reglermöte '98, Preprints*, pages 89–93, Lund, Sweden.

Hagenblad, A. and Ljung, L. (1998). Maximum likelihood identification of Wiener models with a linear regression initialization. In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 712–713, Tampa, Florida, USA.

Hagenblad, A. (1999). Initialization and model reduction for Wiener model identification. In *The 7th Mediterranean Conference on Control and Automation*, pages 716–723, Haifa, Israel.

2

The Estimation Problem

In this chapter, the parameter estimation problem is formulated as a prediction error minimization. General conditions for consistency are applied to the Wiener model. The true predictor may often be difficult to compute exactly for Wiener models. However, under additional assumptions on the noise and the true system, it is shown that an approximative predictor still gives a consistent estimate.

Numerical methods are used to minimize the prediction error criterion. These are reviewed in Section 2.4. An alternative approach is the Expectation Maximization (EM) algorithm, which is outlined in Section 2.5. The chapter ends with an overview of other methods for Wiener model identification described in the literature.

2.1 The Prediction Error Method

A Wiener model with noise is depicted in Figure 2.1. The input u is a known deterministic signal, while the output y is the system output with added measurement noise. The intermediate signal x cannot be measured. v denotes process noise and e measurement noise. We will assume that they are independent of each other and have zero mean. G is a linear system, so noise at the input can be transformed to colored process noise.

q denotes the time shift operator, $qu(t) = u(t+1)$, and θ are the pa-

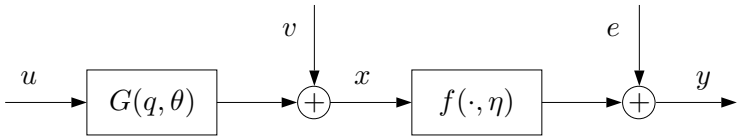


Figure 2.1: The Wiener model. The intermediate signal x is not measurable, while the input u and the output y are measurable, y with noise.

rameters determining the linear dynamic system $G(q,\theta)$. The nonlinear function $f(x,\eta)$ is a function of the output of the dynamic subsystem, and depends also on the parameters η. Different model structures for both G and f are possible, each with their pros and cons. Some issues on different parameterizations are discussed in Chapter 3; here we assume an arbitrary but fixed model structure. The output as described by the Wiener model can be written as

$$y(t) = f\big(G(q,\theta)u(t) + v(t), \eta\big) + e(t) \tag{2.1}$$

For fixed values of the parameters θ and η, this model can be used to predict the output for a given input. The best prediction is the one that gives the conditional expected value of $y(t)$ with respect to the noise v and e, given the values of the old inputs and outputs. We will denote this with $\hat{y}(t,\theta,\eta)$, since it not only depends on time but also on the parameters θ and η. The set of old data up to time t, $\{u(1), y(1), u(2), y(2), \ldots, u(t), y(t)\}$, is denoted by Z^t. The formal definition of the predictor $\hat{y}(t,\theta,\eta)$ is then

$$\hat{y}(t,\theta,\eta) = E\big(y(t)|Z^{t-1},\theta,\eta\big) \tag{2.2}$$

To evaluate the quality of a model, we compare the predicted output $\hat{y}(t,\theta,\eta)$ with the measured output $y(t)$ in the following prediction error criterion:

$$V_N(\theta,\eta) = \frac{1}{N}\sum_{t=1}^{N}\big(y(t) - \hat{y}(t,\theta,\eta)\big)^2 = \frac{1}{N}\sum_{t=1}^{N}\varepsilon^2(t,\theta,\eta) \tag{2.3}$$

where N is the number of measurements. $\varepsilon(t,\theta,\eta)$ is the prediction error. The "closer" the predicted and the measured outputs are, the smaller the criterion $V_N(\theta,\eta)$. The prediction error estimate is the θ and η minimizing $V_N(\theta,\eta)$, denoted as

$$(\hat{\theta},\hat{\eta}) = \arg\min_{(\theta,\eta)} V_N(\theta,\eta) \tag{2.4}$$

2.2 Consistency

Suppose that the measured data y is actually generated according to the following equation:

$$y(t) = f\big(G(q,\theta_0)u(t) + v(t), \eta_0\big) + e(t) \tag{2.5}$$

The corresponding true system is shown in Figure 2.2. A desired property of the parameter estimation method is that if we apply it to data $\{u(t), y(t)\}$ from this system, it should yield the true parameter values θ_0 and η_0. Since our estimate from a particular identification experiment will always be affected by the noise realization at that experiment, this can not be expected in general. Making the thought experiment that we have infinitely many data, we want the effect of the noise on the estimate to be insignificant, and the estimated values of θ and η to be equal to the true values θ_0 and η_0. Such an estimate is said to be consistent.

Definition 1 (Consistency) *Suppose that the true system is described by the parameters θ_0 and η_0. Let θ_N and η_N denote the estimates obtained from a data set with N data. The estimate is said to be* consistent *if $\theta_N \to \theta_0$ and $\eta_N \to \eta_0$ when $N \to \infty$.*

The consistency question is studied in detail in Ljung (1978). Here we will give a brief outline. The development is rather general, and applies in particular to Wiener models.

Let $\widehat{\Theta}_N$ denote the minimizing argument of the prediction error criterion, (2.3),

$$\widehat{\Theta}_N = \arg\min_{\Theta} \frac{1}{N}\sum_{t=1}^{N}\big(y(t) - \hat{y}(t,\Theta)\big)^2 \tag{2.6}$$

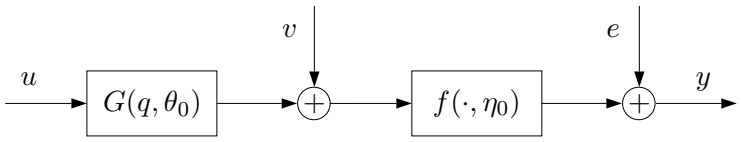


Figure 2.2: The true Wiener system with measurement noise $e(t)$ and process noise $v(t)$.

The consistency question is divided into two parts: We first show that $\widehat{\Theta}_N$ converges to the minimizing argument of a deterministic criterion, then that the true parameters also minimize that criterion.

We will denote $\lim_{N\to\infty}\frac{1}{N}\sum_{t=1}^{N} Ef(\cdot)$ with $\overline{E}f(\cdot)$, and assume that all such sums converge. This is true for a large class of stochastic signals, see Ljung (1999). We will also use the notation

$$\overline{V}(\Theta) = \overline{E}\big(y(t) - \hat{y}(t,\Theta)\big)^2 \tag{2.7}$$

Note that $\overline{V}(\Theta)$ is a deterministic function. We also have that when $N \to \infty$,

$$\widehat{\Theta}_N \to \arg\min_{\Theta} \overline{V}(\Theta) \tag{2.8}$$

Write the system as

$$y(t) = E\big(y(t)|Z^{t-1}\big) + w(t) \tag{2.9}$$

where

$$E\big(w(t)|Z^{t-1}\big) = 0 \tag{2.10}$$

We will denote $E\big(y(t)|Z^{t-1}\big)$ with $\hat{y}_0(t)$.

Inserting (2.9) into the limiting criterion (2.7) we get

$$\begin{aligned}\overline{V}(\Theta) &= \overline{E}\big(\hat{y}_0(t) + w(t) - \hat{y}(t,\Theta)\big)^2 \\ &= \overline{E}\big(\hat{y}_0(t) - \hat{y}(t,\Theta)\big)^2 + \overline{E}w^2(t) + 2\overline{E}w(t)\big(\hat{y}_0(t) - \hat{y}(t,\Theta)\big) \\ &= \overline{E}\big(\hat{y}_0(t) - \hat{y}(t,\Theta)\big)^2 + \overline{E}w^2(t)\end{aligned} \tag{2.11}$$

The term $2\overline{E}w(t)\big(\hat{y}_0(t) - \hat{y}(t,\Theta)\big)$ is zero since $\hat{y}_0(t)$ and $\hat{y}(t,\Theta)$ only depend on old input and output values (cf (2.9) and (2.2)), and $w(t)$ according to the definition (Equation (2.10)) is uncorrelated with old data. The second term is independent of the parameters. $\widehat{\Theta}_N$ will therefore converge to the minimizing argument of the first term.

Now assume that there exists some Θ_0 such that $\hat{y}_0(t) = \hat{y}(t,\Theta_0)$. Then Θ_0 will also minimize $\overline{V}(\Theta)$, since the first term will be zero for $\Theta = \Theta_0$.

This general result can be applied to Wiener models. The output can then be described as

$$y(t) = f\big(G(q,\theta)u(t) + v(t),\eta\big) + e(t) \tag{2.12}$$

To use the prediction error method, we must specify the predictor $\hat{y}(t,\theta,\eta) = E\big(y(t)|Z^{t-1},\theta,\eta\big)$. This is most easily done in the state-space framework. Suppose the linear model is described by the following equation:

$$\xi(t+1) = A(\theta)\xi(t) + B(\theta)u(t) \tag{2.13}$$

$$x(t) = C(\theta)\xi(t) + v(t) \tag{2.14}$$

and that $y(t) = f(x(t),\eta)$. Introducing the new state variable

$$X(t) = \begin{pmatrix} \xi(t) \\ v(t-1) \end{pmatrix} \tag{2.15}$$

the whole Wiener system can be written in state-space form as

$$X(t+1) = \begin{pmatrix} A & 0 \\ 0 & 0 \end{pmatrix} X(t) + \begin{pmatrix} B \\ 0 \end{pmatrix} u(t) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} v(t) \tag{2.16}$$

$$y(t) = f\big(\begin{pmatrix} C & 1 \end{pmatrix} X(t)\big) + e(t) \tag{2.17}$$

For colored process noise $v(t)$, additional states may be needed to describe $v(t)$.

It is difficult to exactly formulate the predictor in this general case, since the process noise $v(t)$ enters nonlinearly in $y(t)$. We may, however, use an approximative approach such as the extended Kalman filter (Anderson and Moore, 1979).

Under some assumptions on the noise and on the true system, we can show consistency for a less complicated, approximative predictor.

1. $e(t)$ is uncorrelated with old data, $E\big(e(t)|Z^{t-1}\big) = 0$. The predictor can then be written as

$$\hat{y}(t,\theta,\eta) = E\Big(f\big(G(q,\theta)u(t) + v(t),\eta\big)\big|Z^{t-1}\Big) \tag{2.18}$$

   This assumption is, e.g., fulfilled if the measurement noise is white.

2. $f\big(G(q,\theta_0)u(t) + v(t),\eta_0\big) = f\big(G(q,\theta_0)u(t),\eta_0\big) + \tilde{f}\big(Z^{t-1},v(t)\big)$, where $E\Big(\tilde{f}\big(Z^{t-1},v(t)\big)|Z^{t-1}\Big) = 0$. In this case the predictor can be simplified to

$$\hat{y}_s(t,\theta,\eta) = E\Big(f\big(G(q,\theta)u(t),\eta\big)|Z^{t-1}\Big) = f\big(G(q,\theta)u(t),\eta\big) \tag{2.19}$$

   This is equivalent to saying that the process noise can be transformed to additive noise on the output, which is uncorrelated with past data. See Figure 2.3.

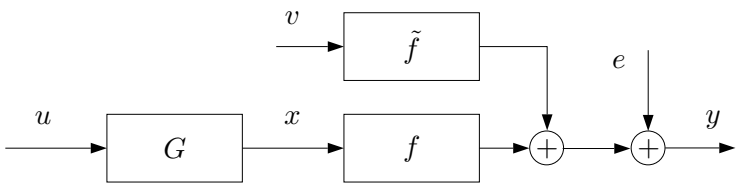


Figure 2.3: A Wiener model subject to Assumption 2. The process noise can be transformed to additive noise on the output, which is uncorrelated with passed data.

Assumption 2 is a strong assumption on the system. It is fulfilled if f is linear (and $v(t)$ is uncorrelated with past data), or if there is no process noise. Other than this, it is in general *not* fulfilled. In this thesis, we still use the simplified predictor (2.19), but we cannot guarantee consistency if Assumption 2 is not fulfilled.

Returning to the characterization (2.11), this is in fact the best we can hope for in a general setting. If all noise variances are zero, this is the error between the true system output and the estimated system output. To be able to conclude that this implies also that the estimated parameters are equal to the true values, we have to impose further conditions both on the input signal $u(t)$, and on the nonlinearity f.

Since the intermediate signal $x(t)$ is not measured, a fixed gain can be arbitrarily distributed between the linear and the nonlinear system. We can never distinguish a linear system $G_0(q)$ and a nonlinearity $f_0(\cdot)$ from a linear system $\alpha G_0(q)$ and a nonlinearity $\alpha^{-1} f_0(\cdot)$. When we discuss consistency in the following, this gain factor is disregarded.

We will state four different conditions. Together, they will be sufficient to ensure that $\hat{y}(t, \Theta) = \hat{y}(t, \Theta_0)$ implies $\Theta = \Theta_0$ (except for a constant gain as mentioned above).

C1. The linear model structure has to be globally identifiable.

C2. The input data has to be informative enough.

C3. The nonlinearity must be invertible.

C4. $\{x(t)\}_{t=1}^{N}$, the input to the nonlinearity, must be dense in the support of f, when N tends to infinity.

Note however that due to the complicated nature of the nonlinear Wiener model, these conditions need not be necessary, and there may be other

sufficient conditions.

Since a linear dynamic system is a special case of the Wiener model, conditions for linear systems must be satisfied. The first two conditions are such conditions. The third condition deals with the coupling between the linear and the nonlinear system, while the last condition is needed to guarantee that the nonlinear system can be identified. We will discuss each condition in turn (see Ljung, 1999, for a more complete treatment in the case of linear systems).

Identifiability (Condition C1)

A linear model structure is a set of stable predictors, $\{\hat{x}(t,\theta)\}$, where θ belongs to some subset of $\mathbf{R}^n$. To be able to get a unique estimate of θ, we must restrict the set of considered linear models, to make sure that different predictors produces different outputs. This is captured in the identifiability concept.

Definition 2 (Identifiability) *A model structure* $\{\hat{x}(t,\theta)\}$ *is* globally identifiable at θ^* *if*

$$\hat{x}(t,\theta) = \hat{x}(t,\theta^*) \quad \Rightarrow \quad \theta = \theta^* \tag{2.20}$$

It is globally identifiable *if it is globally identifiable at almost all* θ^*.

Examples of globally identifiable structures are ARX models, and OE models where the numerator and denominator polynomials have no common factors (see Section 3.1.1 for more on ARX and OE models).

Informative Enough Data Sets (Condition C2)

The actual predictor value for a given parameter value θ depends on the input-output data set $Z^\infty = \{z(t)\}_{t=1}^\infty = \{(u(t), y(t))^T\}_{t=1}^\infty$. For a linear time-invariant model this can be written using a filter $W(q,\theta)$, where

$$\hat{y}(t,\theta) = W(q,\theta)z(t) = \begin{pmatrix} W_u(q,\theta) & W_y(q,\theta) \end{pmatrix} \begin{pmatrix} u(t) \\ y(t) \end{pmatrix} \tag{2.21}$$

It is clear that if, e.g., $u(t) \equiv 0$ we cannot tell much about the system. The information content in the data is addressed in the following definition:

Definition 3 (Informative enough sets) *A data set Z^∞ is* informative enough *if, for two linear time-invariant models $W(q,\theta_1)$ and $W(q,\theta_2)$,*

$$\overline{E}\Big(\big(W(q,\theta_1)-W(q,\theta_2)\big)z(t)\Big)^2 = 0 \tag{2.22}$$

implies that $W(e^{i\omega},\theta_1) = W(e^{i\omega},\theta_2)$ for almost all ω.

An informative enough data set thus allows us to draw the conclusion that two predictors are the same if the mean square difference between the predictors is zero. If the model structure is also identifiable, we have that $\theta_1 = \theta_2$.

The Nonlinearity (Condition C3)

The two conditions discussed above are sufficient in the case of a linear model. In the Wiener model case, the output also goes through a nonlinearity before being measured. If the nonlinearity f is a constant, it is clear that the linear system cannot be identified. If f is invertible, $x(t) = f^{-1}\big(y(t)\big)$ can be computed from $y(t)$. $y(t)$ will then contain the same information as $x(t)$. This is thus a sufficient condition.

It may be possible to identify the linear system also if f is non-invertible. For example, if f is linear in a small region around the origin, the system will be linear for sufficiently small input signals, and thus possible to identify if only the first two conditions are satisfied.

The Input to the Nonlinearity (Condition C4)

To uniquely determine a general nonlinear function f poses several problems. In this chapter we have assumed that f can be parameterized and described by a set of parameters η. How to select a general parameterization is discussed in Section 3.2. Here we assume that we know a priori that f belongs to the parameterized model class. We also assume that f is at least piecewise continuous on some interval $[a,b]$. What we need is sufficiently many data to determine the parameters. A possible condition is then that the input data set is *dense*. Loosely speaking, this means that we have data all over the interval, there are no "holes". A realization of a stochastic variable with a continuous distribution (e.g., a normal or a rectangular distribution) on an interval $[a,b]$ will produce a data set which is dense in that interval. A formal definition is given below.

Definition 4 (Dense data sets) *A data set $E \subset F$ is* dense *in F if for every point $f \in F$ and every given $\varepsilon > 0$ there is a point $e \in E$ such that $|e - f| < \varepsilon$.*

If f is continuous on an interval $[a, b]$ and the input data set $\{x(t)\}$ is dense in that interval, f will thus be uniquely determined by the input-output data $\{x(t), y(t)\}$. This is a general, sufficient, but rather restrictive condition.

In practice it might be unfeasible to have dense input data $x(t)$. One reason is that a dense data set is always infinite, but it is also difficult since we have no direct control over $x(t)$, only via the input to the linear subsystem, $u(t)$. However, if $\{x(t)\}$ is dense in an interval $[a, b]$, we may determine f uniquely in that interval. It is thus important to choose the input signal similar to what is expected when the model is used.

Conclusion

The four conditions discussed above are as said sufficient to guarantee that $\hat{y}(t, \Theta) = \hat{y}(t, \Theta_0)$ implies $\Theta = \Theta_0$, but not necessarily true given this implication. For a given model structure, it is possible to obtain less restrictive conditions.

It is important to note that the minimum of $\overline{V}(\Theta)$ is in fact attained for $\Theta = \Theta_0$. The following chapters discuss how to obtain a good initial estimate of the parameters. If this estimate is close enough to the true parameters, the numerical methods presented in Section 2.4 will converge to the global minimum of the criterion, and under the assumptions made in this section this is obtained for the true parameter values.

2.3 Maximum Likelihood

To obtain the well-known maximum likelihood estimate, we consider the measurements as realizations of stochastic variables. The stochastic variable $Y = \begin{pmatrix} y(1) & y(2) & \dots & y(N) \end{pmatrix}$ then has a probability density function (PDF) $f_Y(\theta, \eta, \mu)$. Here the subscript Y denotes that it is the PDF of the vector-valued stochastic variable Y, while μ denotes the value taken by Y. For a given value of μ, the probability that the observation (measurement) of Y should take that value is proportional to $f_Y(\theta, \eta, \mu)$. The maximum likelihood estimate is the one that maximizes this probability, or the likeli-

hood of the measured value:

$$(\hat{\theta}_{\mathrm{ML}}, \hat{\eta}_{\mathrm{ML}}) = \arg\max_{\theta,\eta} f_Y(\theta, \eta, \mu) \tag{2.23}$$

f_Y is also called the *likelihood function* when we insert a certain value of μ in the probability density function. With some abuse of notation we will often let Y denote both the stochastic variable, and the value of this stochastic variable.

Suppose that the model structure is such that

$$\begin{aligned} \hat{y}(t, \theta, \eta) &= g(t, Z^{t-1}, \theta, \eta) \\ y(t) &= \hat{y}(t, \theta, \eta) + \varepsilon(t, \theta, \eta) \end{aligned} \tag{2.24}$$

where the prediction errors $\varepsilon(t, \theta, \eta)$ are independent and have a probability density function $f_\varepsilon(\epsilon, \theta, \eta)$. The likelihood function for Y is then

$$\begin{aligned} f_Y(\theta, \eta, Y) &= \prod_{t=1}^{N} f_\varepsilon(y(t) - g(t, Z^{t-1}, \theta, \eta), t, \theta, \eta) \\ &= \prod_{t=1}^{N} f_\varepsilon(\varepsilon(t, \theta, \eta), t, \theta, \eta) \end{aligned} \tag{2.25}$$

Maximizing a function is the same as maximizing the logarithm, since the logarithm is a strictly increasing function, so we may instead maximize

$$\log f_Y(\theta, \eta, Y) = \sum_{t=1}^{N} \log f_\varepsilon(\varepsilon(t, \theta, \eta), t, \theta, \eta) \tag{2.26}$$

or minimize

$$-\frac{1}{N} \log f_Y(\theta, \eta, Y) = \frac{1}{N} \sum_{t=1}^{N} -\log f_\varepsilon(\varepsilon(t, \theta, \eta), t, \theta, \eta) \tag{2.27}$$

Note the similarity with Equation (2.3). We now measure the prediction error with the negative log likelihood function instead of a quadratic criterion.

If the prediction errors are not only independent but also Gaussian, each with zero mean and covariance λ, we have

$$-\log f_\varepsilon(\varepsilon, t, \theta, \eta) = \text{const} + \frac{1}{2} \log \lambda + \frac{\varepsilon^2}{2\lambda} \tag{2.28}$$

For a known λ, the first two terms are independent of the parameters θ and η. To minimize (2.27) is then equivalent to minimizing the quadratic criterion (2.3). This means that the maximum likelihood estimate coincides with the estimate minimizing the prediction error criterion.

2.4 Optimization Methods

The numerical methods described in this section can be found in several books on optimization, see, e.g., Dennis and Schnabel (1983), Luenberger (1984) or Ljung (1999).

For a selected model structure and a set of measurements we can now state the criterion function we want to minimize. In general, this is too complex to minimize analytically, so we have to use numerical search methods.

Suppose we have an initial estimate of θ and η, and we want to calculate better estimates, which lowers the value of the prediction error criterion (2.3). The following iterative scheme is often used:

$$\begin{pmatrix}\theta\\ \eta\end{pmatrix}^{(i+1)} = \begin{pmatrix}\theta\\ \eta\end{pmatrix}^{(i)} + \alpha_i h^{(i)} \tag{2.29}$$

$h^{(i)}$ is a search direction and α_i a positive constant used to ensure that the criterion (2.3) is decreased in each iteration step. By selecting the search direction in a proper way we can then guarantee convergence to a local minimum of the criterion.

Since the gradient of a function points in the direction of its steepest ascent, it is natural to base the search direction $h^{(i)}$ on the (negative) gradient of the criterion. For the case of Wiener models, the gradient can be derived analytically, if the nonlinearity f is at least piecewise smooth. Recall that the linear model is parameterized with the parameter vector θ and the nonlinearity with η. The gradient of the criterion is then

$$V_N'(\theta,\eta) = -\frac{1}{N}\sum_{t=1}^{N}\Psi(t,\theta,\eta)\varepsilon(t,\theta,\eta) \tag{2.30}$$

where Ψ denotes the gradient of $\hat{y}$ with respect to θ and η. Using the expression (2.19) for $\hat{y}$ and expanding Ψ and ε we get

$$V_N'(\theta,\eta) = -\frac{1}{N}\sum_{t=1}^{N}\begin{pmatrix} f_x'\big(\hat{x}(t,\theta),\eta\big)\hat{x}_\theta'(t,\theta)\\ f_\eta'\big(\hat{x}(t,\theta),\eta\big)\end{pmatrix}\big(y(t)-\hat{y}(t,\theta,\eta)\big) \tag{2.31}$$

We may note here that if f is the identity function the first row is exactly the same as in the case of estimating a linear system. For given parameters of the linear and nonlinear system, the derivative can be calculated analytically, or a numerical approximation can be used.

If we use the (negative) gradient as search direction, we can guarantee that the prediction error criterion (2.3) decreases in each iteration by selecting a small enough α_i. This is called the *gradient* or the *steepest-descent* method. Note that we do not necessarily need to select the α_i that gives the largest decrease of the criterion, any α_i decreasing the criterion will do. A simple implementation of this might thus start with a large α_i, check if this makes the criterion decrease, otherwise divide α_i with 2 and check again.

To increase the convergence rate, Newton type methods can be used close to the minimum. In the Newton method the gradient is multiplied with the inverse of the Hessian. If the criterion is quadratic in the parameters this guarantees convergence in one step. Close to a local minimum the criterion can be approximated by a quadratic function of the parameters and thus the Newton method has fast convergence there.

The drawback of the Newton method is that it can be costly and/or complicated to calculate the Hessian. The Gauss-Newton method uses the search direction

$$h^{(i)} = \left[H_N\big(\theta^{(i)}, \eta^{(i)}\big)\right]^{-1} V_N'\big(\theta^{(i)}, \eta^{(i)}\big) \tag{2.32}$$

where

$$H_N(\theta, \eta) = \frac{1}{N}\sum_{t=1}^{N} \Psi(t, \theta, \eta)\Psi^T(t, \theta, \eta) \tag{2.33}$$

If the prediction errors are independent the Gauss-Newton search direction is close to the Newton direction.

With the Gauss-Newton method the search direction is thus

$$h^{(i)} = -\left[\frac{1}{N}\sum_{t=1}^{N} \Psi(t, \theta, \eta)\Psi^T(t, \theta, \eta)\right]^{-1} \frac{1}{N}\sum_{t=1}^{N} \Psi(t, \theta, \eta)\varepsilon(t, \theta, \eta) \tag{2.34}$$

This is the least squares solution to the overdetermined system of equations

$$\Psi^T(t, \theta, \eta)h^{(i)} = \varepsilon(t, \theta, \eta), \quad t = 1, 2, \ldots, N \tag{2.35}$$

Equation (2.35) can be solved using QR-factorization, which gives us an efficient way to calculate the Gauss-Newton search direction.

To be able to use these gradient-based methods, the functions involved must be differentiable. For piecewise smooth functions, the differentiability is in general not a problem, since the derivative ceases to exist only in a finite number of points. In the generic case, the probability that we should

end up in one of these points is zero. In practice, the differentiation can be implemented with a difference quotient, which will exist also in the critical points, or calculated analytically. In the latter case, the derivative can be defined as, e.g., zero in the points it ceases to exist. Those points will then not affect the minimization.

2.4.1 Local Minima

A more serious problem is that the prediction error criterion may have several local minima. The numerical search guarantees convergence to one of them, but we cannot be sure that there are no other local minima that gives a lower prediction error. Which minimum the Gauss-Newton search converges to depends on the initial estimate. There are basically two ways to deal with this: either to try several different initial estimates, or to make just one, but make it so accurate that it converges to the global minimum. The former approach can be quite costly, and there are no guarantees that you have found the global minimum. In Chapter 4 we treat the question of finding a good initial estimate.

The problem with local minima is not unique for Wiener models, but well known also for other model structures, like neural networks. Some suggestions on how to initialize the search algorithm can be found in Sjöberg (1997), where it is proposed to start with a linear model, which is then augmented to a nonlinear structure. For Wiener models this would be the approach described in Section 1.3, where the linear system is estimated from input-output data under the assumption that the whole system is linear. We saw in Section 1.3 that there may be problems with this approach.

It should also be noted that the Wiener model is over-parameterized if the linear and nonlinear subsystem are parameterized separately. A constant gain can be distributed arbitrarily between the subsystems, so to get a unique solution, the gain of one subsystem must be fixed. This can be done by expressing the steady-state gain of the linear system as a function of the parameters, and use this as a constraint in the minimization. A simpler solution is to just fix one of the parameters of the linear system, and let it be constant during the minimization. Numerical problems will occur if the over-parameterization is not addressed.

2.4.2 On Linear Regression

If $\hat{y}(t)$, the prediction of the output, is a linear combination of known entities, like the given data Z^{t-1}, the parameter estimate (2.4) is especially

easy to calculate. Suppose that

$$\hat{y}(t) = a_1 x_1(t) + \cdots + a_n x_n(t) \tag{2.36}$$

where $x_i(t)$ may be old inputs or outputs, or given transformations of these. Collecting the data and parameters in vectors, $\varphi(t) = (x_1(t) \dots x_n(t))^T$ and $\Theta = (a_1 \dots a_n)^T$, we may write this as

$$y(t) = \varphi(t)^T \Theta \tag{2.37}$$

The prediction error criterion (2.3) is then

$$V(\Theta) = \frac{1}{N} \sum_{t=1}^{N} \big(y(t) - \varphi(t)^T \Theta\big)^2 \tag{2.38}$$

This criterion can be minimized explicitly by setting the gradient of $V(\Theta)$ equal to zero:

$$-2\frac{1}{N} \sum_{t=1}^{N} \varphi(t)\big(y(t) - \varphi(t)^T \Theta\big) = 0 \tag{2.39}$$

which has the solution

$$\Theta = \left[\frac{1}{N} \sum_{t=1}^{N} \varphi(t)\varphi(t)^T\right]^{-1} \frac{1}{N} \sum_{t=1}^{N} \varphi(t) y(t) \tag{2.40}$$

The inverse does not have to be calculated explicitly in practice, but QR-factorization can be used. This gives a numerically better and more stable method (Dennis and Schnabel, 1983).

The minimization of the quadratic criterion (2.38) is known as *linear regression*. Three things makes this especially interesting: It is possible to solve the problem analytically, there are fast numerical methods to compute the solution, and the solution of the problem is unique. This is all due to the fact that the parameters enter linearly in the predictor (2.36). In Chapter 4 we will show how the Wiener system can be parameterized with parameters that enter linearly. This will allow us to use linear regression to obtain an estimate which is unique and fast to calculate.

2.4.3 The Instrumental Variables Method

Suppose that there exist some true values for a_i, such that the data y and x_i satisfy the equation

$$y(t) = a_1^0 x_1(t) + \cdots + a_n^0 x_n(t) + e(t) = \varphi(t)^T \Theta^0 + e(t) \tag{2.41}$$

where $e(t)$ is noise. Inserting this into Equation (2.40) for the linear regression estimate, we see that the linear regression estimate will be

$$\Theta = \Theta^0 + \left[\frac{1}{N}\sum_{t=1}^{N}\varphi(t)\varphi(t)^T\right]^{-1}\frac{1}{N}\sum_{t=1}^{N}\varphi(t)e(t) \tag{2.42}$$

provided that the inverse exists. When the number of data, N, tends to infinity, the estimate will equal the true parameter values if the last sum is zero; that is, if the noise $e(t)$ is uncorrelated with the regressor vector $\varphi(t)$. The estimate will thus be consistent.

If the noise and the regressor vector are correlated, we may still obtain a consistent estimate by modifying Equation (2.39). Instead of the regressor vector, we use the so-called *instruments* or *instrumental variables* $\zeta(t)$. The instrumental-variable or IV estimate is the solution to the equation

$$\frac{1}{N}\sum_{t=1}^{N}\zeta(t)\big(y(t) - \varphi(t)^T\Theta\big) = 0 \tag{2.43}$$

To get consistency we have to choose the instruments $\zeta(t)$ such that

$$\frac{1}{N}\sum\zeta(t)\varphi(t)^T \tag{2.44}$$

is nonsingular, and $\frac{1}{N}\sum\zeta(t)e(t)$ tends to zero as N tends to infinity. The IV method is described in detail (including consistency constraints) in Ljung (1999).

2.5 The Expectation Maximization Algorithm

An alternative approach to the parameter estimation problem in Wiener model identification is the Expectation Maximization (EM) algorithm. An early paper on the EM algorithm is Baum et al. (1970). The algorithm is also described in the survey paper Dempster et al. (1977). Bergman (1998) presents an application to segmentation. We will here describe how it can be applied to Wiener model identification.

Consider again the Wiener model with both process noise and measurement noise depicted in Figure 2.4. We will adopt a stochastic framework, and introduce the following notations: $X = \big(x(1) \quad \dots \quad x(N)\big)^T$ and $Y = \big(y(1) \quad \dots \quad y(N)\big)^T$ are stochastic, vector valued variables. $u(t)$ is a deterministic (known) input signal. $v(t)$ and $e(t)$ are both stochastic processes. $\Theta = (\theta, \eta)$ are the parameters we want to estimate. The maximum

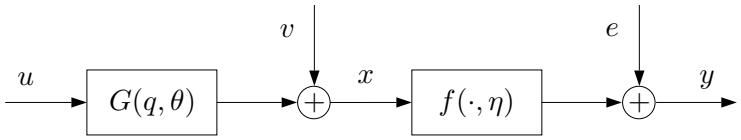


Figure 2.4: The Wiener model with process and measurement noise.

likelihood estimate maximizes the log likelihood of Y given Θ. We will denote this likelihood with $p(Y|\Theta)$.

The idea behind the EM algorithm is that the likelihood $p(Y|\Theta)$ may be hard to derive, but it would be easier if we also had measurements of some other variable, X. We have that

$$p(X, Y|\Theta) = p(X|Y, \Theta)p(Y|\Theta) \tag{2.45}$$

or equivalently

$$\log p(Y|\Theta) = \log p(X, Y|\Theta) - \log p(X|Y, \Theta) \tag{2.46}$$

Since the left hand side is independent of X, we may multiply both sides with a function $f(X)$ and integrate, if we choose f such that $\int f(X)\, dX = 1$. In particular, we may choose $f(X) = p(X|Y, \Theta')$, the conditional PDF for X given Y and $\Theta = \Theta'$:

$$\begin{aligned}\log p(Y|\Theta) &= \int \log p(X, Y|\Theta) p(X|Y, \Theta')\, dx - \int \log p(X|Y, \Theta) p(X|Y, \Theta')\, dx \\ &= E\big(\log p(X, Y|\Theta)|Y, \Theta'\big) - E\big(\log p(X|Y, \Theta)|Y, \Theta'\big) \\ &= Q(\Theta, \Theta') - H(\Theta, \Theta')\end{aligned} \tag{2.47}$$

The EM algorithm is defined as follows:

The EM algorithm

Alternate the following two steps:

1. Compute the conditional mean of $\log p(X, Y|\Theta)$ given Y and $\Theta = \Theta_p$ for a fix Θ_p.

$$Q(\Theta, \Theta_p) = E\big(\log p(X, Y|\Theta)|Y, \Theta_p\big) \tag{2.48}$$

2. Determine Θ_{p+1} as

$$\Theta_{p+1} = \arg\max_{\Theta} Q(\Theta, \Theta_p) \tag{2.49}$$

It can be shown that for each pass of the EM algorithm, the log likelihood $\log p(Y|\Theta)$ increases. This relies on the fact that $H(\Theta, \Theta') \leq H(\Theta', \Theta')$. This is easy to show using Jensen's inequality. See, e.g., Dempster et al. (1977).

The EM algorithm is potentially interesting when identifying Wiener models, since it can be described as using averaging over data that is not readily available. In the Wiener model case, this is the intermediate signal $x(t)$. We will now apply the EM algorithm to our Wiener model identification problem. Y is the output, X is the unmeasurable intermediate signal. Θ is the parameters, θ of the linear system and η of the nonlinear.

First we need an expression for $\log p(X, Y|\Theta)$. Using Baye's rule, we have

$$p(X, Y|\Theta) = p(Y|X, \Theta)p(X|\Theta) = p(Y|X, \eta)p(X|\theta) \tag{2.50}$$

so since X does not depend on η

$$\log p(X, Y|\Theta) = \log p(Y|X, \eta) + \log p(X|\theta) \tag{2.51}$$

For the first term on the right hand side, $\log p(Y|X, \eta)$, we have that $y(t) = f\big(x(t), \eta\big) + e(t)$. If X and η are known, the only stochastic part is $e(t)$. We assume that the measurement noise $e(t)$ is white and Gaussian, with zero mean and variance σ^2. Since $f(x)$ is a static nonlinearity, the output $y(t)$ will also be an independent sequence, and

$$p(Y|X, \eta) = \prod_{t=1}^{N} p\big(y(t)|x(t), \eta\big) = \prod_{t=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{\big(y(t) - f(x(t), \eta)\big)^2}{2\sigma^2} \right\}$$

which gives

$$\log p(Y|X, \eta) = -\frac{1}{2} \sum_{t=1}^{N} \log 2\pi\sigma^2 - \frac{1}{2\sigma^2} \sum_{t=1}^{N} \big(y(t) - f(x(t), \eta)\big)^2 \tag{2.52}$$

The first term can be disregarded with respect to the EM algorithm, since it is independent of the parameters Θ.

Now we turn to the second factor of Equation (2.50), $p(X|\theta)$. We have that

$$x(t) = G(q,\theta)u(t) + v(t) \tag{2.53}$$

$u(t)$ is a deterministic signal and θ is assumed known. If $v(t)$ is Gaussian, X given θ will also be Gaussian, with a mean m_θ and a covariance matrix P_θ. We then have

$$p(X|\theta) = \frac{1}{\sqrt{(2\pi)^N \det P_\theta}} \exp\Big\{-\frac{1}{2}(X - m_\theta)^T P_\theta^{-1}(X - m_\theta)\Big\}$$

so

$$\log p(X|\theta) = -\frac{N}{2}\log 2\pi - \frac{1}{2}\log\det P_\theta - \frac{1}{2}(X - m_\theta)^T P_\theta^{-1}(X - m_\theta) \tag{2.54}$$

Also here the first term is independent of Θ and can be disregarded.

We can now form $Q(\Theta, \Theta_p)$ by taking the expectation of Equations (2.52) and (2.54) with respect to X, for given $\Theta = \Theta_p$ and Y. Note that in doing this, the parameters θ and η in (2.52) and (2.54) are free. Since (2.52) depends only on η and (2.54) only on θ, we will split $Q(\Theta, \Theta_p)$ in two parts,

$$Q(\Theta, \Theta_p) = Q_1(\eta, \Theta_p) + Q_2(\theta, \Theta_p) \tag{2.55}$$

where $Q_1(\eta, \Theta_p)$ is the part coming from (2.52) and $Q_2(\theta, \Theta_p)$ the part from (2.54). Disregarding the terms that are independent of Θ and X, and the factor $1/2$ which is common to all terms, we then obtain

$$Q_1(\eta, \Theta_p) = -\frac{1}{\sigma^2}\sum_{t=1}^{N} E\big(y(t) - f(x(t),\eta)\big)^2 \tag{2.56}$$

$$Q_2(\theta, \Theta_p) = -\log\det P_\theta - E\big((X - m_\theta)^T P_\theta^{-1}(X - m_\theta)\big) \tag{2.57}$$

or equivalently, after applying the expectation to each term and then completing the squares,

$$Q_1(\eta, \Theta_p) = -\frac{1}{\sigma^2}\sum_{t=1}^{N}\Big\{\big(y(t) - Ef(x(t),\eta)\big)^2 + Ef^2(x(t),\eta) - E^2 f(x(t),\eta)\Big\} \tag{2.58}$$

$$Q_2(\theta, \Theta_p) = -\log\det P_\theta - (EX - m_\theta)^T P_\theta^{-1}(EX - m_\theta) - \operatorname{tr}\big(P_\theta^{-1}(EXX^T - EXEX^T)\big) \tag{2.59}$$

We have used that $X^T AX = \mathrm{tr}(AXX^T)$, and that we may exchange the order of the expectation and the trace operators. All expectations are with respect to X, given $\Theta = \Theta_p$ and Y.

Equation (2.59), $Q_2(\theta, \Theta_p)$, can be interpreted in a smoothing framework: $E(X|Y, \Theta_p)$ is the best estimate of X given our data Y and the model parameters Θ_p. $EXX^T - EXEX^T$ is the corresponding variance. These quantities can be found by using the fixed-interval smoothing variant of the Kalman filter. Since the output $y(t)$ is a nonlinear function of $x(t)$ we will need the extended Kalman filter, EKF. A standard reference on Kalman filtering is Anderson and Moore (1979). With both process noise $v(t)$ and measurement noise $e(t)$, this problem is comparable to the problem of formulating the predictor (2.2), discussed on page 13.

The first line in Equation (2.58) is even harder to calculate. For a given structure of the nonlinear system, we may expand the expressions for $Ef(x(t), \eta)$ and $Ef^2(x(t), \eta)$, but the calculations will be tedious and the expressions complicated. Note however that if the nonlinearity is known, this step can be excluded.

The conclusion of the EM algorithm applied to Wiener models is thus that no obvious advantages are visible, when compared to the prediction error method. When using the prediction error method, it is hard to express the true predictor for the output $y(t)$, and we have to approximate it with an extended Kalman filter. Using the EM algorithm, we instead have to find the predictor for $x(t)$ given y, which also has to be approximated with an extended Kalman Filter. The two methods share the problem with local minima, and the need for a good initial estimate.

2.6 Other Approaches

Several approaches to the identification of Wiener models have been proposed. Many of them can be included in the general prediction error framework we have presented. The authors have chosen different parameterizations that each have their advantages and disadvantages. Some have restricted the input to have special properties. We will give short summaries of some approaches here.

Wigren (1993) derives a recursive prediction error algorithm. He describes the linear block with a transfer function operator, and the nonlinear block as piecewise linear. The x-space is partitioned into segments, where the slope and bias of the function are the parameters. The partition points are however supposed to be fixed. Conditions for local convergence to the

true parameter vectors are stated, and the method is applied to simulated data from a control valve. In Wigren (1994), the nonlinearity is assumed known, and conditions both for local and global convergence are shown.

Pajunen (1992) treats the problem of model-reference adaptive control of a Wiener system. The linear system is represented as a transfer function, and the inverse of the nonlinearity is represented with linear B-splines. The method is applied to a pH control process.

Several papers treat the case where the input is (white) Gaussian noise. Bussgang's theorem (Bussgang, 1952) states that the cross-correlation of two Gaussian signals, where one of them has undergone a nonlinear transformation, is proportional to the cross-correlation before the transformation. This can be applied to Wiener systems where the input is Gaussian.

Hunter and Korenberg (1986) use Gaussian input and estimates the linear and nonlinear subsystems iteratively. Several examples of biological systems that can be described as Wiener models are cited. The linear system is estimated from the cross-correlation function, and the nonlinear system is described with a polynomial.

Billings and Fakhouri (1977, 1982) uses the notion of separable processes, which is related to Bussgang's theorem. This is possible to apply to white Gaussian input signals. The impulse response of the linear system can then be estimated from the cross-correlation function between the input and the output. The nonlinearity is described with a power series. Also other block oriented structures than Wiener models are considered.

Greblicki (1994) also assumes white Gaussian input and disturbances, and observes that the inverse of the nonlinearity is then proportional to the expected value of the input in the previous time step, given the output at a certain time. This is used to estimate the nonlinearity as an orthogonal series. Convergence under some assumptions is proved, and an algorithm to identify also the linear system is proposed.

Subspace methods are used to identify the linear subsystem in Westwick and Verhaegen (1996), and it is shown that if the input is (colored) Gaussian, this estimate is consistent. The basic algorithm is developed for odd nonlinearities, i.e., $f(-x) = -f(x)$, but it is also extended to general nonlinearities. The nonlinearity is expressed as a power series.

In Bruls et al. (1997), a state space model is used for the linear system and Chebyshev polynomials for the nonlinearity. This makes it possible to phrase the prediction error minimization as a separable least squares problem, which has better numerical properties than the original problem. The numerical search is initialized with a subspace estimate of the linear

system as in Westwick and Verhaegen (1996).

Related to the separable least squares method is the method proposed in Zhu (1998). The Hammerstein model, where the static nonlinear block comes before the linear block, is discussed. The nonlinearity is parameterized with polynomials and the linear block with a high order ARX model. The prediction error criterion is then bilinear in the parameters, which means that it can be solved as iterated least-squares problems. The high order ARX model is reduced using a frequency-domain criterion. In Zhu (1999b) the method is extended to Wiener models.

Kalafatis et al. (1997) select a parameterization where all parameters enter linearly, and can thus minimize a quadratic error criterion explicitly. FIR or FSF (frequency sampling filters) are suggested for the linear subsystem, and a power series or B-splines for the nonlinearity. The same method is treated in Kalafatis et al. (1995), where it is applied to a pH process.

There are two different fundamental difficulties in these approaches. With a flexible model structure we risk getting stuck in a local minimum. A less flexible structure may reduce or eliminate the local minima, but need a larger number of parameters to produce an accurate estimate. More parameters to estimate might also demand more data. A good initial estimate will reduce the risk of ending up in a local minimum.

Our approach is to combine the good parts in both these approaches. First use a model structure with many parameters that enter linearly in the problem and make an estimate. Then use this estimate as an initial estimate in the iterative numerical search to minimize the prediction error criterion for a more flexible structure. In Chapter 3 different possible parameterizations are discussed, and in Chapter 4 a particular parameterization is chosen that enables a linear regression estimate of the parameters. Chapter 5 describes the model reduction necessary to proceed with the prediction error minimization. The complete algorithm, and an analysis of it, is presented in Chapter 6.

3

Parameterizations

In this chapter we discuss different ways of parameterizing the Wiener model. To stress the block structure of the model we will parameterize the two subsystems independently. We will treat the linear block in Section 3.1 and the nonlinear block in Section 3.2.

3.1 The Linear Block

The linear block is a system which is linear, dynamic, time invariant, causal, and stable. We will assume it is represented in discrete time (continuous time systems are not treated in this thesis). Such a system G is completely described by its impulse response $\{g(t)\}_{t=1}^{\infty}$. For a given input $u(t)$ we have the output $x(t)$ as follows:

$$x(t) = \sum_{k=1}^{\infty} g(k)u(t-k) \quad t = 0, 1, 2, ... \tag{3.1}$$

The *transfer function* $G(q)$ of the system is

$$G(q) = \sum_{k=1}^{\infty} g(k)q^{-k} \tag{3.2}$$

and the system output can then be written as $x(t) = G(q)u(t)$. If $G(q)$ is stable, we have that

$$\sum_{k=1}^{\infty} |g(k)| < \infty \tag{3.3}$$

Although a system is uniquely determined by its impulse response, it is impractical to work with this in general infinite sequence. Instead we want an expression where the system G is characterized by a finite number of parameters. We will collect the parameters into a parameter vector θ, and write $G = G(q, \theta)$. Rational transfer functions, state space models and FIR models are some possible parameterizations that we will discuss. In (3.1) we have assumed that there is a time delay of one time unit, and no direct term from $u(t)$ to $x(t)$ in the linear system. This is a natural assumption, and implies no loss of generality since the input can always be time shifted to make sure we have a unit delay. It is also possible to explicitely express a time delay of n_k samples.

3.1.1 Rational Transfer Functions

A common choice is to select the transfer function as a rational function where the numerator and denominator coefficients are the parameters.

$$G(q, \theta) = \frac{b_1 q^{-1} + \cdots + b_{n_b} q^{-n_b}}{1 + a_1 q^{-1} + \cdots + a_{n_a} q^{-n_a}} = \frac{B(q)}{A(q)} \tag{3.4}$$

or equivalently

$$x(t) + a_1 x(t-1) + \cdots + a_{n_a} x(t-n_a) = b_1 u(t-1) + \cdots + b_{n_b} u(t-n_b) \tag{3.5}$$

The parameter vector is

$$\theta = \begin{pmatrix} a_1 & \dots & a_{n_a} & b_1 & \dots & b_{n_b} \end{pmatrix}^T \tag{3.6}$$

In a real life situation, we always have noise, and we may wish to model the color of this noise. Depending on how we model the noise, the rational transfer function presented here can be described as an ARX model, an output error (OE) model or a Box-Jenkins model. In the ARX (Auto-Regressive with eXogeneous input) model, the white noise $e(t)$ enters directly into the difference equation:

$$\begin{aligned} x(t) + a_1 x(t-1) + \cdots + a_{n_a} x(t-n_a) = \\ = b_1 u(t-1) + \cdots + b_{n_b} u(t-n_b) + e(t) \end{aligned} \tag{3.7}$$

In the OE model we instead assume that the noise is added to the output of the transfer function:

$$x(t) = \frac{B(q)}{A(q)}u(t) + e(t) \tag{3.8}$$

This is a natural description if the disturbance is white measurement noise. We may also separately model the color of the noise added to the output, and will then get a Box-Jenkins (BJ) model:

$$x(t) = \frac{B(q)}{A(q)}u(t) + \frac{C(q)}{D(q)}e(t) \tag{3.9}$$

Like $A(q)$ and $B(q)$, $C(q)$ and $D(q)$ are polynomials in q^{-1}.

3.1.2 Finite Impulse Response Models

A special case of the rational transfer function model is the finite impulse response, FIR, model. We here assume $A(q) = 1$, which gives the following difference equation:

$$x(t) = b_1 u(t-1) + b_2 u(t-2) + \cdots + b_{n_b} u(t-n_b) \tag{3.10}$$

An FIR model of order n_b can only describe systems whose impulse response has maximum length of n_b time steps, but if we let n_b tend to infinity, any given (stable) system will be possible to describe accurately. The expression power of the FIR model thus only depends on how many parameters we are willing to use to describe it.

Another important feature of the FIR model is that the output is a linear function of the parameters. If we want to find the parameters that minimize a quadratic error criterion as in Chapter 2, we can then use linear regression to calculate the minimum explicitely.

3.1.3 Laguerre and Kautz Models

A drawback when using FIR models is that we need a large number of parameters to describe a system with a slow impulse response, or a poorly damped system. Alternative representations, where prior knowledge about the dominant poles can be utilized, are the Laguerre and Kautz models (Wahlberg, 1991, 1994; Lindskog, 1996).

The Laguerre model describes the transfer function $G(q,\theta)$ with the following basis function expansion:

$$G(q,\theta) = \sum_{k=1}^{N_L} \bar{g}_k L_k(q,a) \tag{3.11}$$

$$\text{where} \quad L_k(q,a) = \frac{\sqrt{1-a^2}}{q-a}\left(\frac{1-aq}{q-a}\right)^{k-1} \tag{3.12}$$

θ denotes the parameters $\bar{g}_k$ while $a \in \mathbf{R}$ is a filter coefficient chosen a priori. It can be shown (Wahlberg, 1991) that if a is close to the true pole of the system, the number of parameters N_L needed to describe the system with a given accuracy is in general much smaller than n_b, the number of FIR parameters needed.

If the system has resonant (complex) poles, the Laguerre model may still need many parameters. A more general structure is then the Kautz model, which corresponds to the Laguerre filter when a is allowed to be complex. The Laguerre basis functions $L_k(q,a)$ are then replaced with the Kautz functions $\Psi_k(q,b,c)$ where

$$\Psi_{2l-1}(q,b,c) = \frac{\sqrt{1-c^2}\,(q-b)}{q^2+b(c-1)q-c}\left(\frac{-cq^2+b(c-1)q+1}{q^2+b(c-1)q-c}\right)^{l-1} \tag{3.13}$$

$$\Psi_{2l}(q,b,c) = \frac{\sqrt{(1-c^2)(1-b^2)}}{q^2+b(c-1)q-c}\left(\frac{-cq^2+b(c-1)q+1}{q^2+b(c-1)q-c}\right)^{l-1} \tag{3.14}$$

b and c should be chosen so that the roots of $q^2+b(c-1)q-c=0$ are close to the true poles of the system. In that case only a small number of parameters will be needed to describe the system.

It is also possible to extend the Laguerre/Kautz filters to include several a priori known poles. See Wahlberg (1994); Lindskog (1996).

3.1.4 Linear State Space Models

Using a state space model, the relation between the input and the output is the following:

$$\begin{aligned} \xi(t+1) &= A\xi(t) + Bu(t) \\ x(t) &= C\xi(t) \end{aligned} \tag{3.15}$$

ξ is the state vector, consisting of n state variables. A, B and C are matrices, of dimension $n \times n$, $n \times 1$ and $1 \times n$, respectively. This representation is equivalent to a transfer function representation with $n_a = n$ and $n_b = n-1$. Bruls et al. (1997); Westwick and Verhaegen (1996) use a state space representation of the linear block in their treatment of Wiener models.

3.1.5 The Frequency Sampling Filter

The frequency sampling filter can be seen as a linear transformation of the FIR model. The following equations define the frequency sampling filter (FSF):

$$\begin{aligned} x(t) &= \sum_{k=-(n-1)/2}^{(n-1)/2} G(e^{i\omega_k}) f_k(t) \\ f_k(t) &= H_k(q)u(t) = \frac{1}{n}\frac{1-q^{-n}}{1-e^{i\omega_k}q^{-1}}u(t) \\ \omega_k &= 2\pi k/n \end{aligned} \tag{3.16}$$

$G(e^{i\omega_k})$ are the parameters and also the discrete frequency response of the system at ω_k.

Since the FSF model is a linear transformation of the FIR model, it has the same expression power. It also has the characteristic that the output is a linear function of the parameters. It has been argued (see Kalafatis et al., 1997) that the FSF allows fewer parameters than the FIR model in certain cases.

3.2 The Nonlinear Block

Next, we consider the static nonlinear block. We will denote it by f, which we will also use for the function realized in the block. f is thus a real-valued function of one variable, with input x and output y (since the block is static we omit the time index), $y = f(x)$.

To express the nonlinear function, we will use a function expansion with basis functions and parameters. This is not a new idea, so many different possibilities have been suggested. The main structure is the following:

$$y = \sum_{i=1}^{n_b} f_i B_i(x) \tag{3.17}$$

For our purpose, we will distinguish two different types: those with fixed or somehow predetermined basis functions B_i, where f_i are the only parameters, and those where $B_i(x) = B_i(x, \eta)$ also contain parameters η which can affect for example the shape and position of the basis function. The first case may be considered a special case of the second, where the parameters contained in B_i are fixed beforehand.

If the internal parameters of the basis functions B_i are fixed, the output is a linear function of the parameters. This allows us to use linear regression, as described in Section 2.4.2, to estimate the parameters. The two basic advantages with linear regression is that it is fast and gives a unique estimate. In the general case, a prediction error criterion may have several local minima, and we need time-consuming numerical methods to minimize the criterion. Internal parameters in B_i will in general give more flexibility, but the parameter estimation problem will be harder.

In the following presentation of some possible choices of basis functions, x is assumed to be scalar. It is however often possible to generalize the ideas to the vector-valued case.

3.2.1 Power Series

In a power series, the basis functions are simply powers of x.

$$B_i(x) = x^i \quad i = 0, 1, 2, \dots n_b \tag{3.18}$$

It is well known (see Dahlquist and Björk, 1974) that a power series have good approximation abilities close to a fix (selected) point. The drawback of using them is that the approximation far from the selected point may be poor. The approximating function will also often show oscillatory behavior.

The power series basis functions are independent of the parameters, so linear regression can be used to estimate the parameters from data.

3.2.2 Chebyshev Polynomials

If polynomial approximation of the nonlinearity is desirable, Chebyshev polynomials might be a better choice than a power series. The Chebyshev polynomial basis functions are defined as follows

$$B_k(x) = \cos(k \arccos x) \tag{3.19}$$

No parameters are contained in the basis function. Chebyshev polynomials are interesting because they have the so-called minimax property: Among

all polynomial of degree n, where the coefficient for x^n is equal to one, the Chebyshev polynomials have the smallest maximum norm on $[-1, 1]$. This will in general make an approximation by Chebyshev polynomials less oscillative than a power series. More properties of Chebyshev polynomials can be found in Dahlquist and Björk (1974). Chebyshev polynomials are used to describe the nonlinearity in Bruls et al. (1997).

3.2.3 B-splines

Splines are also known as piecewise polynomials: A number of polynomials, joined together at so-called *breakpoints*, or *knots*. A first order spline is just a piecewise constant function. A second order spline is a continuous, piecewise linear function. If the polynomial pieces are of order n (and the spline of order $n + 1$) they can be connected to form a function with a continuous $n - 1$ order derivative. For example, a third order spline consists of quadratic pieces, which are joined together at the breakpoints to make the first derivative of the function continuous. We will concentrate on (piecewise) linear splines.

Splines can be expressed in different ways. One convenient way of expressing the spline as a basis function expansion, fitting in our framework, is the B-form. For n given breakpoints $\{x_i\}_{i=1}^n$, $B_i(x)$ is the unique piecewise linear function satisfying:

$$B_i(x_j) = \begin{cases} 0 & \text{if } i \neq j \\ 1 & \text{if } i = j \end{cases} \tag{3.20}$$

Some basis functions are depicted in Figure 3.1.

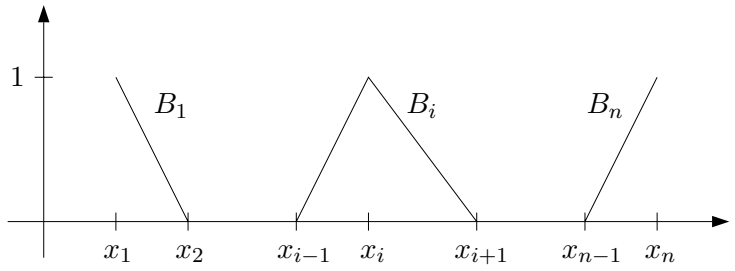


Figure 3.1: B-spline basis functions of order 1 (piecewise linear).

We can also explicitely state the equation of a B-spline basis function:

$$B_i(x) = \begin{cases} 0 & \text{if} \quad x < x_{i-1} \\ \frac{x-x_{i-1}}{x_i-x_{i-1}} & \text{if} \quad x_{i-1} \leq x < x_i \\ \frac{x_{i+1}-x}{x_{i+1}-x_i} & \text{if} \quad x_i \leq x < x_{i+1} \\ 0 & \text{if} \quad x_{i+1} \leq x \end{cases} \tag{3.21}$$

If the breakpoints are selected/determined in advance, all the parameters enter linearly in the B-splines representation. The selection of the breakpoints is of course a delicate matter. They do not need to be evenly spaced. Normally it is desirable to have many breakpoints where the function is changing rapidly, but if the function is unknown this is not easy to specify. One way is to let the breakpoints be parameters to estimate too. This will make the structure more flexible, but make the parameter estimation harder.

We might note that for a continuously differentiable function on a closed bounded interval, we can obtain a B-spline approximation of arbitrary accuracy by using sufficiently many breakpoints. The proof is related to the proof that every continuous function can be arbitrarily well approximated by piecewise constants, used in basic textbooks in calculus, see e.g. Rudin (1976).

Proposition 3.1 *Suppose f is a continuously differentiable function on the closed bounded interval $[a, b]$. Then for every $\varepsilon > 0$ there exists a piecewise linear continuous function g such that*

$$|f(x) - g(x)| < \varepsilon \quad \textit{for all } x \in [a, b] \tag{3.22}$$

The proposition holds also on non-bounded intervals, if the derivative of f is uniformly continuous on that interval.

A good reference treating splines is de Boor (1978). de Boor has also written a spline toolbox for MATLAB, (de Boor, 1992).

3.2.4 Neural Networks

The concept of neural networks is based on the use of basis functions, the most common one being the sigmoid. The sigmoid basis function has the following equation:

$$B_k(x) = \frac{1}{1 + e^{-(\eta_{0k} + x\eta_{1k})}} \tag{3.23}$$

The basis function has two internal parameters. η_{1i} determines how fast the transition from 0 to 1 is, and η_{0i} the position of the transition. We may note that for $\eta_{1i} > 0$, the sigmoid tends to zero as x tends to minus infinity, and to one as x goes to plus infinity; for $\eta_{0i} + x\eta_{1i} = 0$ its value is 0.5.

As with splines, it is possible to show that "nice" functions can be approximated arbitrarily well with sigmoids (Cybenko, 1989). The sigmoid can be considered as a smooth version of a piecewise constant basis function which can take only one of two values: 1 or 0, on or off.

Contrary to splines, there are no breakpoints we need to select a priori, this is handled by the internal parameters η_{0i} and η_{1i}. On the other hand, this makes the parameter estimation harder; it is well known that a quadratic error criterion as in Chapter 2 has several local minima for neural networks, which is a problem. We can only guarantee convergence to *some* local minimum. See Haykin (1994) for more on neural networks.

3.2.5 Hinging Hyperplanes

Hinging hyperplanes were introduced in Breiman (1993). We will use the parameterization proposed in Pucar and Sjöberg (1996).

Hinging hyperplanes use basis functions called *hinge functions*. A hinge function in one dimension has the following equation:

$$B_0(x) = \eta_{00} + \eta_{10}x \tag{3.24}$$

$$B_i(x) = \begin{cases} \eta_{0i} + \eta_{1i}x & \text{if } x > -\frac{\eta_{0i}}{\eta_{1i}} \\ 0 & \text{otherwise} \end{cases} \tag{3.25}$$

$$i = 1, 2, \ldots, M$$

Figure 3.2 shows an example of a hinge function. Since the hinge function is piecewise linear, a sum of hinge functions will be a piecewise linear function. The parameters η_{0i} and η_{1i} determines the slope of the linear pieces and the breakpoints.

Any piecewise linear function can be expressed as a sum of hinge functions. Since linear B-splines according to Proposition 3.1 are able to approximate any continuously differentiable function arbitrarily well, this is also true for hinging hyperplanes. The difference between hinging hyperplanes and B-splines is that we consider the breakpoints fixed a priori for B-splines. Apart from that, B-splines and hinging hyperplanes are just two different ways of expressing piecewise linear functions. Switching between the two representations is thus possible.

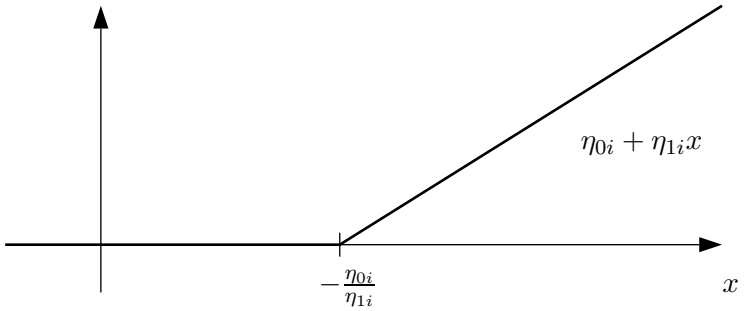


Figure 3.2: A hinge function

Since there are parameters inside B_i, we will have the same problems with numerical minimization of the quadratic prediction error criterion, and local minima as for neural networks.

3.2.6 Wavelets

The wavelet basis functions consists of scaled and dilated versions of a "mother wavelet" ψ. Often two indices i and j are used to parameterize the basis functions, so a basis function can be written

$$B_{i,j}(x) = \frac{1}{\sqrt{2^i}}\psi\Big(\frac{x - 2^i j}{2^i}\Big) \tag{3.26}$$

The i, or rather 2^i, is a scale parameter and the j a dilatation. It can be shown (see Mallat, 1998) that $\{B_{i,j}\}_{(i,j)\in\mathbb{Z}^2}$ is an orthonormal basis of $\mathbf{L^2}(\mathbb{R})$, if the wavelet function ψ satisfies certain conditions. Two examples of mother wavelets ψ are given here. For more on wavelets see Mallat (1998).

Haar Wavelets

The Haar wavelets are piecewise constant functions, described by the following equation

$$\psi(t) = \begin{cases} -1 & \text{if } 0 \le t < \frac{1}{2} \\ 1 & \text{if } \frac{1}{2} \le t < 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.27}$$

Since the wavelet is piecewise constant, the approximating function will also be piecewise constant.

Shannon Wavelets

The Shannon wavelet is described by the equation

$$\psi(t) = \frac{\sin 2\pi(t-1/2)}{2\pi(t-1/2)} - \frac{\sin \pi(t-1/2)}{\pi(t-1/2)} \tag{3.28}$$

The Shannon and the Haar wavelets are shown in Figure 3.3.

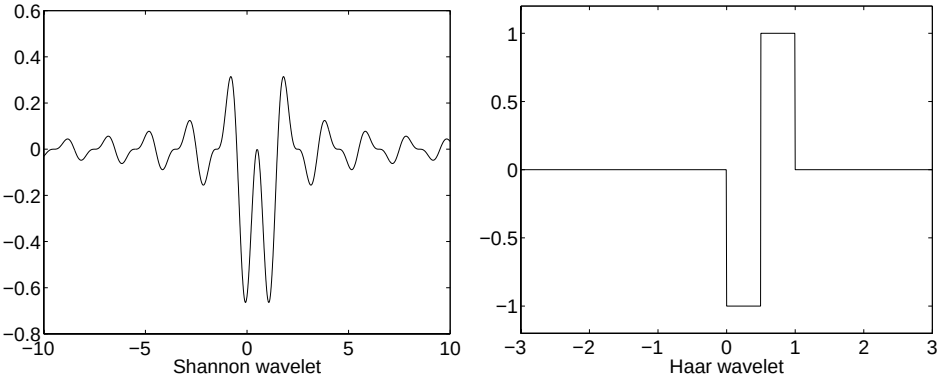


Figure 3.3: Examples of wavelet basis functions. Shannon wavelet to the left, Haar wavelet to the right

4

The Initial Estimate

The estimation problem formulated in Chapter 2 can be solved using a numerical search method if we have a good initial estimate. This chapter shows how a unique initial estimate can be obtained in a fast and numerically reliable way. The initial estimate relies on a particular parameterization of the Wiener model, where the parameters enter linearly. We can then use linear regression to estimate the parameters.

4.1 An Initial Estimate via Linear Regression

Recall the Wiener model depicted in Figure 4.1.

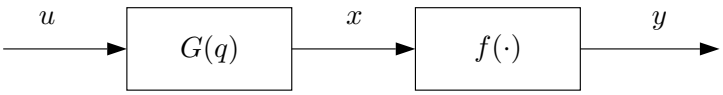


Figure 4.1: The Wiener model

We will derive the initial estimate using the following assumptions:

1. The linear subsystem $G(q)$ is stable.
2. The nonlinear function f is invertible.

3. There is no noise in the system.

This will make it easier to develop the estimate, and relaxations of the second and third assumptions are discussed further on. Unstable linear systems are not treated in this thesis.

Common nonlinearities which are not invertible are for example a deadzone and a saturation. It is then not possible to show convergence theoretically, but the initial estimate can still be used, even if no guarantees can be given for its usefulness. Convergence properties of the estimate when noise is present are discussed in Chapter 6. It is of course the normal situation that we have noise, but to simplify the discussion in this chapter we exclude it.

We can write the intermediate signal $x(t)$ as a linear function of the parameters if we use an FIR model for G.

$$x(t) = b_1 u(t-1) + b_2 u(t-2) + \cdots + b_{n_b} u(t-n_b) \tag{4.1}$$

It was noted in Section 3.1.2 that by selecting n_b large enough, an FIR model can always describe a stable G accurately enough. We then parameterize the *inverse* of the nonlinear system, f^{-1}, with linear B-splines, and use that $x(t) = f^{-1}\big(y(t)\big)$:

$$x(t) = \sum_{i=1}^{n_f} f_i B_i\big(y(t)\big) \tag{4.2}$$

Linear B-splines were described in Section 3.2.3, where it was also noted that the selection of the breakpoints can be a delicate matter. However, we also know according to Proposition 3.1 that with a large number of tightly enough spaced breakpoints we can approximate f^{-1} arbitrarily well.

Putting together Equations (4.1) and (4.2) we get:

$$\sum_{i=1}^{n_b} b_i u(t-i) - \sum_{i=1}^{n_f} f_i B_i(y(t)) = 0 \tag{4.3}$$

This is an equation where the parameters enter linearly and we can thus estimate the parameters with linear regression. We have the parameter vector

$$\begin{pmatrix} \theta^T & \eta^T \end{pmatrix} = \begin{pmatrix} b_1 & \dots & b_{n_b} & f_1 & \dots & f_{n_f} \end{pmatrix} \tag{4.4}$$

and the regression vector

$$\varphi(t)^T = \begin{pmatrix} u(t-1) & \dots & u(t-n_b) & -B_1\big(y(t)\big) & \dots & -B_{n_f}\big(y(t)\big) \end{pmatrix} \tag{4.5}$$

We can write the criterion as

$$\min \sum_{t=1}^{N} \left(\varphi(t)^T \begin{pmatrix} \theta \\ \eta \end{pmatrix} \right)^2 \tag{4.6}$$

Equation (4.6) can not be attacked directly in the linear regression framework since we want to avoid the trivial solution $\theta = \eta = 0$. The criterion measures the error between the output of the linear subsystem, and the output of the inverse of the nonlinearity. If both these are identically zero, the error will be zero and the criterion minimized.

There are several ways to ensure that the trivial solution is banned. Due to the over-parameterization (see Section 2.4), no generality is lost if we fix one parameter, say $b_1 \equiv 1$. We can then write

$$-u(t-1) = \tilde{\varphi}(t)^T \begin{pmatrix} \tilde{\theta} \\ \eta \end{pmatrix} \tag{4.7}$$

where $\tilde{\varphi}$ and $\tilde{\theta}$ are φ and θ without the first element, respectively. This is the standard linear regression situation.

Another possible constraint to use is the following norm constraint:

$$\theta^T\theta + \eta^T\eta = 1 \tag{4.8}$$

This approach is called total least squares, TLS. The TLS problem is treated in Van Huffel and Vanderwalle (1991) and can be solved with singular value decomposition.

A constraint with a natural physical interpretation is

$$\sum_{i=1}^{n_b} b_i = 1 \tag{4.9}$$

This can be interpreted as requiring the static gain of the linear model to be 1. The problem is no longer a linear regression problem, but since the criterion is quadratic and the constraint convex, we still have a unique minimum, and the problem can be solved using quadratic programming. See Luenberger (1984).

To conclude: By parameterizing the linear system with an FIR model, and the inverse of the nonlinear system with linear splines, we get an equation where the parameters enter linearly. By fixing one of the parameters to a constant value, we can use linear regression to estimate the other parameters. We lose no generality in doing this. Another possibility is to calculate

the TLS solution which yields a solution where the norm of the parameter vector is one. We can also fix the static gain of the linear system, and use quadratic programming to get the solution. Either way, we can in a relatively short time get a unique estimate of the system which is the minimum of a quadratic error criterion.

4.2 The Initial Estimate in Practice

When using the initial estimate in practice, additional questions arise. The theorems guaranteeing that our system can be accurately described by the model are valid asymptotically when n_b and n_f tends to infinity. This means we might need many parameters to describe our system, and we do not beforehand know how many. Luckily, the cost for using more parameters is not too large in terms of time when we use the linear regression estimate. The limited number of data available in practice does, however, put a limit of the number of parameters we can estimate. Trying to estimate too many parameters may cause numerical problems, and the estimates are more influenced by noise if we have few data.

Another limitation is that we have assumed f to be invertible. This is essential since we explicitely parameterize the inverse f^{-1}. But even if this condition is violated and the estimate of the nonlinear system is not well defined, the estimate of the linear system may still be interesting. This is illustrated in the example in Section 4.2.2.

We have used that all parameters enter linearly in linear B-splines if the breakpoints are fixed. But how should the breakpoints be fixed in a practical example? Generally speaking, we want more breakpoints where the function is changing rapidly, but before we have estimated the function we do not know where that is. A reasonable approach is to use evenly spread breakpoints in terms of the data point support, that is, in regions where we have many measured output values, we put more breakpoints. Another approach is to space the breakpoints evenly on the interval of the output. A drawback with this method is that we might end up with breakpoints in regions where we have no or few data points, making it harder to obtain an accurate estimate of the corresponding B-spline coefficients. Too few data points in the support of a basis function may give numerical problems. In Chapter 5 on model reduction we discuss possible ways to reduce the number of breakpoints for a given estimate of the nonlinearity.

We will now show some examples of initial estimates from data. All these examples models the linear system with an FIR model, and the inverse of

the nonlinearity with linear B-splines. We try to exploit some of the user's choices, such as the number of parameters, what parameter to fix, and how to select the breakpoints.

4.2.1 A First Example

We start by repeating the motivating example from Section 1.3, and discussing the user's choices made there. Recall that the system was

$$x(t) = \frac{q^{-1}}{(1 - \alpha q^{-1})^2} u(t) \tag{4.10}$$

$$y(t) = e^{x(t)} \tag{4.11}$$

with $\alpha = 0.7$. The input signal is a sum of sinusoids:

$$u(t) = \sum_{k=1}^{20} \sin(k\pi t/10 + \phi_k) \tag{4.12}$$

where ϕ_k is a stochastic variable with uniform distribution on $[0, 2\pi]$. 300 data points were generated, with no added noise.

To visualize the estimates we will use the estimate of the linear system to simulate $x(t)$, and then plot $x(t)$ versus $y(t)$. If the linear system estimate is accurate this will produce a clear picture of the nonlinearity. If the estimate is less accurate, the points will be more scattered around the nonlinearity.

First, we need to select n_b, the number of FIR parameters, and n_f, the number of B-splines. There are no easy ways to do this, but prior knowledge about the system can be of help. More parameters allows us to capture more of the system's behavior, but also require that more data is available to make a reasonable estimate.

If data comes from a sampled system, the sampling interval T should be chosen to reflect the time constant of the physical system. A rule-of-thumb (Ljung, 1999) is to chose the sampling frequency ten times the bandwidth of the system. For systems that are not too oscillative, $n_b = 20$ may then be a good starting point. This holds however only for the dominant pole of the system. If the system has several poles with different time constants, a much larger number of FIR parameters may be needed. If prior knowledge about the positions of the poles is available, Laguerre/Kautz filters may be an interesting alternative.

The number of B-splines, n_f, can be selected relative to n_b. Roughly speaking, selecting $n_f = n_b$ means that we use an equal number of parameters to describe the linear and the nonlinear system. If we want to put more

emphasis on the linear system we should use more parameters for that, and thus select n_b larger than n_f.

If the estimate of the linear system is accurate, the nonlinearity is easy to find by plotting simulated $x(t)$ versus measured $y(t)$. This means we may want to use more parameters for the linear system than for the nonlinearity. We have found that $n_b = 2n_f$ is a good starting point.

In Figure 4.2 the initial estimates are plotted as described above for different choices of n_b, with $n_f = n_b/2$. The breakpoints are marked with stars. It is clear that 10 FIR parameters and 5 breakpoints are too few, the data points are much closer to the curve with 20 FIR parameters and 10 breakpoints.

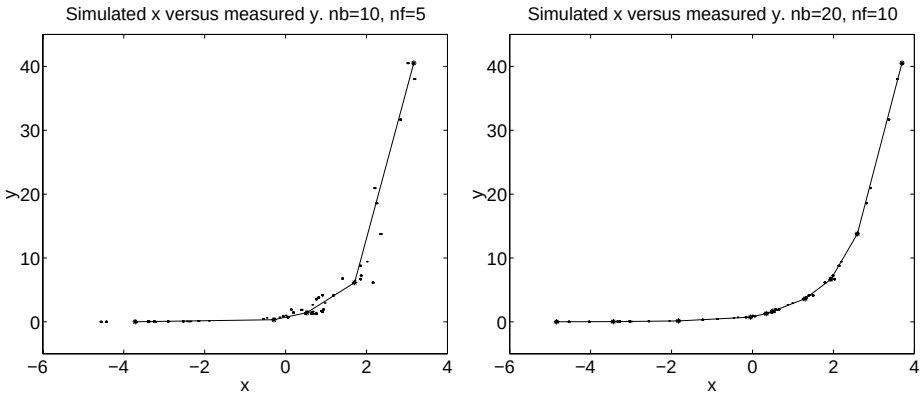


Figure 4.2: Effect of the number of parameters on the initial estimate. Simulated x are plotted against measured y. The solid line shows the estimate of the nonlinearity. Left: $n_b = 10, n_f = 5$. Right: $n_b = 20, n_f = 10$.

The positions of the breakpoints also have to be fixed. Since we have no prior information about where the nonlinearity is changing rapidly (i.e., where we want many breakpoints), a reasonable heuristic is to have many breakpoints where the output data is clustered (this method was used in Figure 4.2). An automatic breakpoint selection procedure may translate this into selecting the breakpoints such that all have equal support from the data. The minimum and maximum output value should also be included in the set of breakpoints. This approach is illustrated to the left in Figure 4.3.

An alternative approach is to just spread the breakpoints evenly between the minimum and maximum output value. Care must then be taken to ensure that the breakpoints have sufficient support. To the right in Fig-

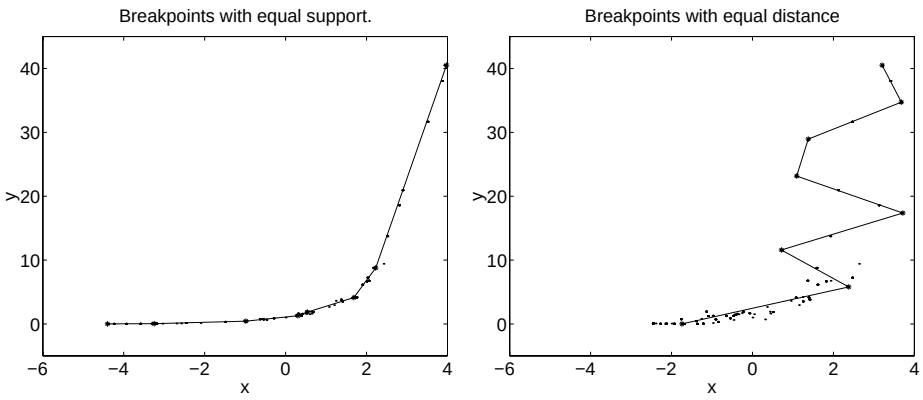


Figure 4.3: Initial estimates with different choices of breakpoints. Simulated x are plotted against measured y. The solid line shows the estimate of the nonlinearity. Left: Breakpoints with equal support from data. Right: Breakpoints with equal distance between the minimum and maximum output value. $n_b = 20, n_f = 8$.

ure 4.3, most of the data only affect the first three breakpoints. The other breakpoints have support only from one or two data points. This gives us a very bad estimate of the nonlinearity, we have even lost the invertibility property.

A third user parameter is how to avoid the trivial solution that all parameters equal zero. Several possibilities were suggested in the last section: To fix a parameter to a constant, to fix the static gain, or to use TLS. For small noise levels, the methods will often give similar results[1], except for a constant gain that may be moved from the linear to the nonlinear subsystem. Higher noise levels may however affect the estimate, since the methods have different numerical properties. A discussion on the accuracy of the different possibilities can be found in Section 6.3. The estimates obtained when fixing the static gain to one and using TLS, respectively, are shown in Figure 4.4. The general shape of the nonlinearity is the same in both plots, but the scale on the x-axis is different.

The computation time will depend on which method chosen. The TLS method requires a singular value decomposition and will typically take longer time than the other two methods. A comparison of typical computation times for this example is listed below. The computation was performed

[1] If the system can be exactly described by the initial model, the results are the same for noise free data.

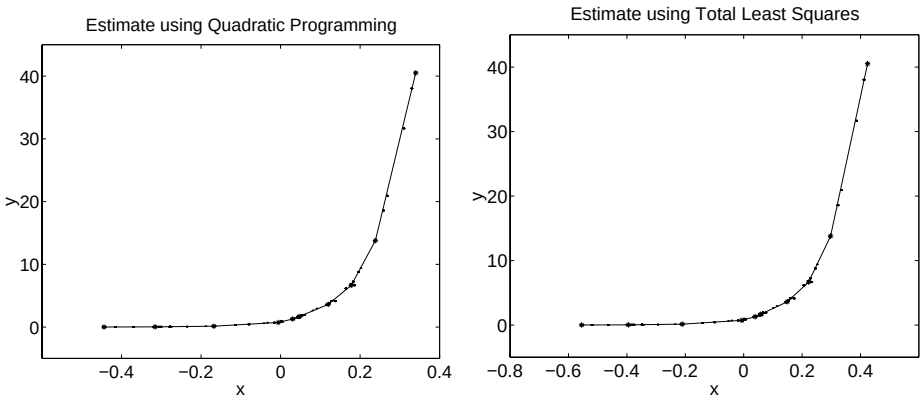


Figure 4.4: Initial estimate using quadratic programming and total least squares, respectively. The breakpoints have equal support from data. $n_b = 20, n_f = 10$. As before, simulated x are plotted against measured y. The solid line shows the estimate of the nonlinearity.

in MATLAB using a Sun Ultra 1/170E.

LR (fixing one parameter to 1):	0.0586 s
QP (fixing the steady-state gain of the linear system to 1):	0.0599 s
TLS (fixing the norm of the parameter vector to 1):	0.3799 s

4.2.2 An Example of a Non-Invertible Nonlinearity

The estimation method we have presented assumes that the nonlinearity is invertible. Here we also show an example of a non-invertible nonlinearity. The system we consider is the following:

$$x(t) = \frac{q^{-1}}{1 - 0.7q^{-1}} u(t) \tag{4.13}$$

$$y(t) = \begin{cases} -0.1x(t) - 1.1 & \text{for } x(t) < -1 \\ x(t) & \text{for } -1 \leq x(t) < 1 \\ -0.1x(t) + 1.1 & \text{for } 1 \leq x(t) \end{cases} \tag{4.14}$$

The nonlinearity is shown to the left in Figure 4.5. The input signal was white Gaussian noise with variance 1.

30 FIR parameters and 20 B-splines were used when calculating the initial estimate. A plot of the simulated x versus the measured y is shown

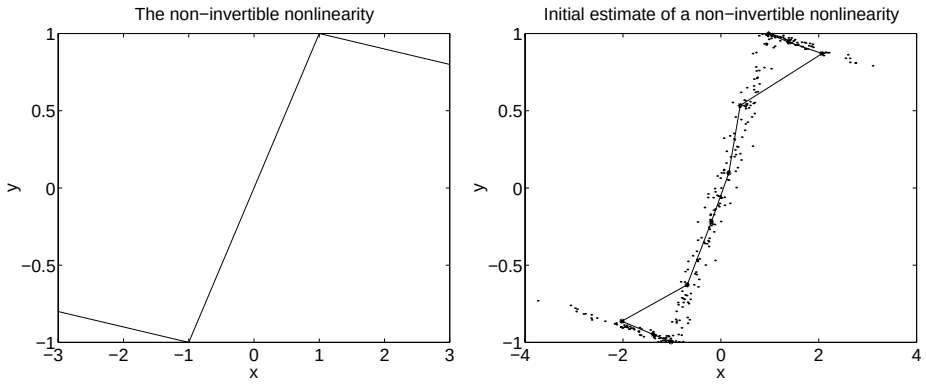


Figure 4.5: Initial estimate of a Wiener model where the nonlinearity is not invertible. Simulated x is plotted versus measured y. Note that even though the estimated nonlinearity (solid line) is not very accurate, the data points are centered around the true nonlinearity.

in Figure 4.5. The estimate of the nonlinearity is not very good, in fact it is not even invertible. A better estimate can however be obtained directly from the plot. In this case, the estimate of the linear system is useful, even if the estimate of the nonlinearity is disregarded. We will return to this example in Section 7.2.

4.2.3 An Example of a System with Noise

We again consider the first example,

$$x(t) = \frac{q^{-1}}{(1-\alpha q^{-1})^2} u(t) \tag{4.15}$$

$$y(t) = e^{x(t)} \tag{4.16}$$

with $\alpha = 0.7$. The input signal is a sum of sinusoids:

$$u(t) = \sum_{k=1}^{20} \sin(k\pi t/10 + \phi_k) \tag{4.17}$$

where ϕ_k is a stochastic variable with rectangular distribution.

Again we generate 300 data points, but this time with added noise. We assume that we can only measure $y(t)$ with noise,

$$y_m(t) = y(t) + e(t) \tag{4.18}$$

where $e(t)$ is independent identically distributed Gaussian noise with variance σ^2. Two different estimates are illustrated in Figure 4.6: One with $\sigma^2 = 0.1$, one with $\sigma^2 = 1$.

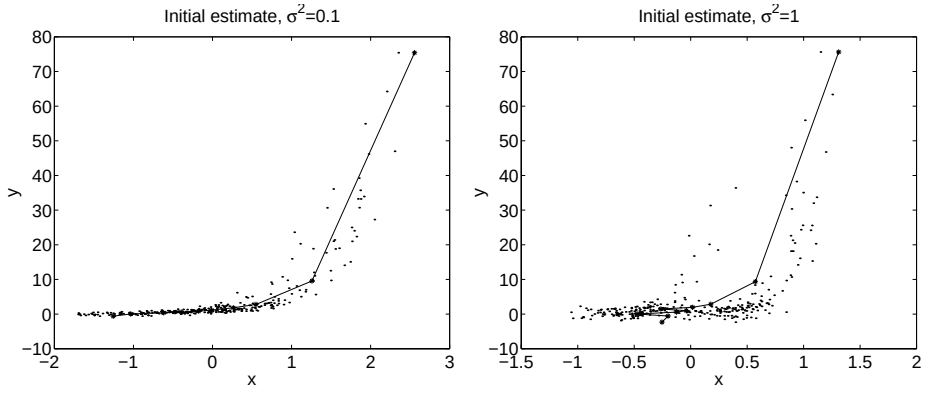


Figure 4.6: Initial estimates from data with measurement noise. Simulated x are plotted against measured y. The solid line shows the estimate of the nonlinearity. The left figure has measurement noise with $\sigma^2 = 0.1$, in the right figure the measurement noise has variance $\sigma^2 = 1$

The first example shows that the method may work well also when we have a small measurement noise. The data points are somewhat scattered around the estimated nonlinearity, but the estimate seems reasonable. In the second example the data points are even more scattered, and the resulting estimate is not invertible any more. We may choose to still use this estimate, or we may use the plot as a guideline to choose other breakpoints. We can see that there are more breakpoints than seem to be needed from $-10 < y < 10$. Selecting the breakpoints to 5, 10, 20, 50 and the minimum and maximum value of y gives us the estimate depicted in Figure 4.7. The data points are still very scattered, but the estimate is invertible, and the shape is close to an exponential.

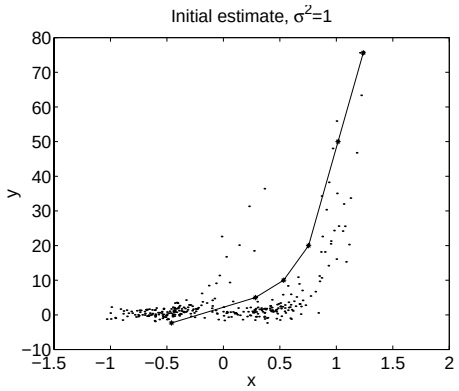


Figure 4.7: Initial estimates with measurement noise ($\sigma^2 = 1$). Simulated x are plotted against measured y. The solid line shows the estimate of the nonlinearity. The breakpoints are here selected to 5, 10, 20, 50 and the minimum and maximum value of y.

5

Model Reduction

We have now shown how to calculate the parameter estimates that minimize the prediction error criterion from an initial estimate (Chapter 2) and how to use a particular parameterization to calculate the initial estimate in a simple and fast, yet efficient way (Chapter 4). To get an accurate estimate, a large number of parameters had to be used. A final estimate with fewer parameters may be preferred, and also a particular model structure of those presented in Chapter 3. Using a large number of parameters is also costly in the Gauss-Newton search.

This chapter presents some possibilities for converting the FIR and spline models to other representations. The overview is not complete, but some attributes and aspects of different methods are discussed. References for the model reduction of the linear subsystem are Zhou et al. (1995), Wahlberg (1989) and Al-Saggaf and Franklin (1988). For the spline breakpoint reduction see de Boor (1978).

5.1 Model Reduction of the Linear System

Starting with an n-th order model G, the model reduction problem consists of finding an r-th order model G_r, which minimizes the error between G and G_r. Different methods for model reduction measure the error in different norms. We may use a prediction error criterion in the time domain, as

described in Chapter 2, or a frequency domain norm, e.g., the 1-norm or the H_∞-norm (see Zhou et al., 1995).

Using the H_∞ norm, balanced truncation aims at minimizing the following error:

$$||G - G_r||_\infty \quad \text{where} \quad ||G||_\infty = \sup_\omega \bar{\sigma}(G(i\omega)) \tag{5.1}$$

$\bar{\sigma}(G(i\omega))$ denotes the largest singular value of the transfer matrix $G(i\omega)$. In the scalar case (single input, single output), this is just the absolute value of the transfer function, $|G(i\omega)|$.

To minimize (5.1), assume that the n-th order model G is a continuous-time state space model.

$$\begin{aligned} \dot{\xi}(t) &= A\xi(t) + Bu(t) \\ x(t) &= C\xi(t) \end{aligned} \tag{5.2}$$

The influence of the different states on the input-output behavior can be expressed with the controllability and observability Gramians R and S, defined by the equations

$$ARA^T + BB^T - R = 0 \tag{5.3}$$

$$A^T SA + C^T C - S = 0 \tag{5.4}$$

A state space transformation will change the Gramians, but not the input-output behavior. The *balanced realization* of the system has $R = S = \Sigma$, where Σ will be a diagonal matrix, with the hankel singular values $\sigma_i > 0$ on the diagonal. It can be shown that a system can always be transformed into a balanced realization (see Zhou et al., 1995). Supposing that the singular values are ordered, $\sigma_1 > \sigma_2 > \cdots > \sigma_n$, we may eliminate the states corresponding to the smallest singular values to approximate the system with a system of lower degree. This is known as *balanced truncation*.

If G is the original model and G_r the reduced model of order r obtained by the balanced truncation, it can be shown that the following infinity norm bound on the error holds:

$$||G - G_r||_\infty \leq 2(\sigma_{r+1} + \cdots + \sigma_n) \tag{5.5}$$

A lower bound can be obtained by using the more complicated Hankel norm approximation. See Zhou et al. (1995); Al-Saggaf and Franklin (1988) for more on model reduction via balanced truncation.

A discrete-time FIR model can always be described by a state-space representation, e.g., using observable or controllable canonical form. The discrete time state-space model can be converted to continuous time, or methods similar to balanced truncation can be applied directly to the discrete time model. Al-Saggaf and Franklin (1988) also discuss the discrete case.

The original linear model G can also be used to simulate noise free output $x(t)$ from the input $u(t)$. The input-output data set $\{u(t), x(t)\}_{t=1}^{N}$ can then be used to identify the lower order model. If $\hat{x}_r(t)$ is the output from the reduced order model, this corresponds to minimizing a quadratic error criterion like

$$\frac{1}{N}\sum_{t=1}^{N}(x(t)-\hat{x}_r(t))^2 \tag{5.6}$$

With G_r a parametric model, this is the linear identification problem, where the part of the dynamics which cannot be modeled by the reduced order model, is considered as noise. In the examples in this thesis, an ARX model is used, but other model structures are also possible.

A variant of this is to consider system identification with a frequency domain error criterion (see, e.g., Wahlberg, 1989). Frequency weighting can be used both in the frequency domain criterion, and in the time domain as a prefilter which is applied to the data.

5.2 Model Reduction of the Nonlinear System

The initialization method presented in Chapter 4 gives us a linear spline approximation of the inverse of the nonlinear system. Provided the estimate of the inverse is invertible, a piecewise linear approximation of the nonlinearity is easily obtained. We have shown that in the noise free case (the case with noise is treated in Chapter 6), we get arbitrarily accurate approximation if using sufficiently many basis functions and enough data, but there is no guarantee that the estimated model based on a particular realization of the data will be invertible.

Can we make the estimated function invertible? One possibility is to use this as a constraint in the calculation of the initial estimate. Using quadratic programming, constraints of the form $f_i \geq f_{i+1} + \epsilon$ are easily incorporated for a given ϵ. With f_i the spline parameters used in Chapter 4, this will enforce invertibility of the estimated nonlinearity. Another possibility is

to use some kind of smoothing. Noise in the measurements will typically produce estimates that are not very smooth.

We will now concentrate on how to reduce the number of breakpoints of the piecewise linear estimate. This is a problem investigated by the spline community, and we have used the `newnot` algorithm suggested by de Boor (1978, 1992). We will present it for the case of linear splines, the generalization to higher order splines can be found in the book de Boor (1978).

Suppose we want to approximate a given function f with a linear function g on the interval $[a, b]$. The following interpolation theorem (de Boor, 1978) is then useful.

Theorem 1 *Let f be two times continuously differentiable on $[a, b]$, and let g be the linear function with $g(a) = f(a)$ and $g(b) = f(b)$. Then*

$$|f(x) - g(x)| \leq \frac{\max |f''|}{2} |b - a|^2 \tag{5.7}$$

For a proof of the theorem see de Boor (1978).

de Boor also gives the following bounds relating the error between an interpolating function g and the function f, with the error between the best (in the maximum-norm sense) approximation h and f:

$$\|f - h\|_{[a,b]} \leq \|f - g\|_{[a,b]} \leq 2\|f - h\|_{[a,b]} \tag{5.8}$$

Using this inequality we have that

$$\|f - h\|_{[a,b]} \leq \frac{1}{2} \|f''\|_{[a,b]} |b - a|^2 \tag{5.9}$$

(We use the notation $\|f\|_I$ to denote the maximum of the function f on the interval I). For a piecewise linear function h, which is linear on the intervals $[x_i, x_{i+1}]$, this means that

$$\|f - h\|_{[a,b]} \leq \max_i \frac{1}{2} \|f''\|_{[x_i,x_{i+1}]} |x_{i+1} - x_i|^2 \tag{5.10}$$

To get the best approximation, we thus want to minimize the right hand side. The minimum is obtained if

$$\|f''\|_{[x_i,x_{i+1}]} |x_{i+1} - x_i|^2 = \text{constant for } i = 1, 2, \ldots, n \tag{5.11}$$

This is equivalent to determining x_i such that $\sqrt{\|f''\|_{[x_i,x_{i+1}]}}|x_{i+1} - x_i|$ is constant, or asymptotically

$$\int_{x_i}^{x_{i+1}} \sqrt{|f''(x)|}\, dx = \text{constant} \tag{5.12}$$

de Boor then uses the variation of f' to construct f''. The command `newknt` in the Spline Toolbox of MATLAB (1996) calculates these new breakpoints.

The use of the second derivative to determine the best breakpoints is also justified by the following heuristic argument: Suppose the true nonlinearity is indeed piecewise linear. The derivative of a piecewise linear function is piecewise constant, so we need few breakpoints in regions where f' is constant or almost constant, and more breakpoints where f' is varying. We therefore consider f'' to find the variation of f'. The `newnot` algorithm assigns an equal amount of variation to each interval.

Other approaches to reduce the number of breakpoints are also possible. Hamann and Chen (1994) makes local approximations of the curve to select the most significant points. Since only a few points is used in each approximation, the method as stated is sensitive to noise. In Schumaker and Stanley (1996) a method preserving properties like monotonicity and convexity is proposed. Numerical examples indicate that the method works also on noisy data. The method reduces breakpoints from quadratic splines, and it is not clear if it is applicable to linear splines.

Given a spline estimate of the inverse nonlinearity with a large number of breakpoints we can thus compute the "best" distribution of a smaller number of breakpoints, and as stated in Section 3.2.5, we can always convert a spline function to hinging hyperplanes. Since also the breakpoints are parameterized in a hinging hyperplane model, we have a good chance of obtaining the optimal breakpoints after minimizing the prediction error criterion (2.3) even if they are slightly wrong after the reduction.

What we do not know is the optimal number of breakpoints. For a given data set, more breakpoints will give a model with smaller error, but the estimate of each parameter will depend on fewer data points, and thus be more dependent on the noise. Fewer breakpoints will reduce the influence of the noise since more data points are used to estimate each parameter. We have a trade off between bias and variance. A classical way to solve this is to use a criterion which takes this into account. Two well-known criteria were formulated by Akaike; they both weight the prediction error criterion $V_N(\theta, \eta)$ with a term that depends on the dimension of the parameter vector.

With $d_{\mathcal{M}} = \dim\theta + \dim\eta$, Akaike's information criterion (AIC) is

$$AIC = V_N(\theta, \eta) + \frac{d_{\mathcal{M}}}{N} \tag{5.13}$$

and Akaike's final prediction error (FPE) criterion is

$$\frac{1 + d_{\mathcal{M}}/N}{1 - d_{\mathcal{M}}/N} V_N(\theta, \eta) \tag{5.14}$$

We refer to Ljung (1999) for more on these criteria.

To convert the initial nonlinearity estimate to another model structure than a piecewise linear one as hinging hyperplanes is not as straightforward. This is equivalent to approximating a given function in a certain model structure, and is a subject of research in approximation theory. A straightforward approach is to use the initial estimate to simulate inputs and outputs of the nonlinearity and estimate the desired model from these. See Braess (1986) for details on approximation theory.

5.2.1 Application of `newnot`

To illustrate how `newnot` selects points, here is a small example. The following nonlinearity was used:

$$y = \begin{cases} 0.1x - 0.9 & \text{for } x < -1 \\ x & \text{for } -1 \leq x < 1 \\ 0.1x + 0.9 & \text{for } 1 \leq x \end{cases} \tag{5.15}$$

A piecewise linear spline with 20 breakpoints was estimated from (x, y) data. The result is shown to the left in Figure 5.1. When eliminating breakpoints, we want to keep the breakpoints that are close to the "corners" of the nonlinearity. The plot to the right in Figure 5.1 shows that this is the case. We have kept six breakpoints.

In a noise free case like this, it is easy to reduce the breakpoints. Instead of using `newnot` we may also use visual inspection to select the relevant breakpoints. In Figure 5.2, the splines are estimated from noisy data. It is then not as clear which breakpoints to choose. We see however that the `newnot` algorithm gives reasonable estimates.

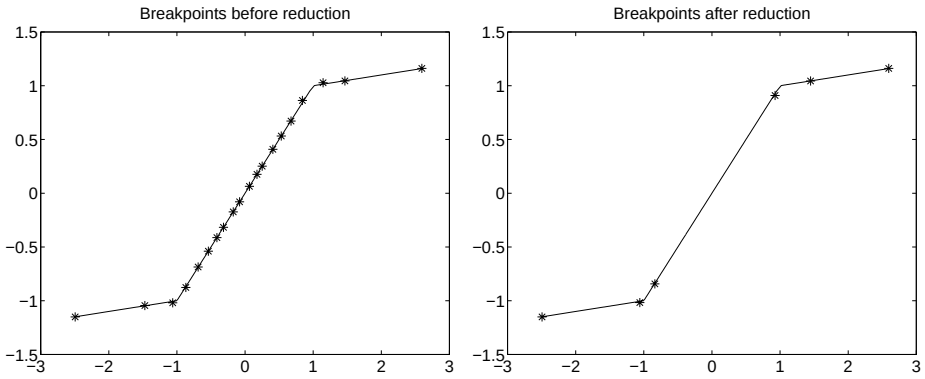


Figure 5.1: Breakpoint reduction example. Original spline to the left, reduced spline to the right. The stars denote breakpoints, the solid line shows the true nonlinearity.

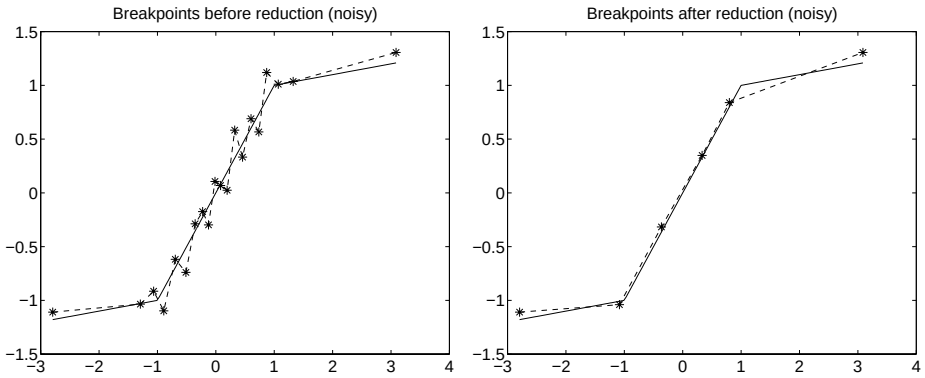


Figure 5.2: Breakpoint reduction with noise. The stars denote breakpoints, the dashed line the approximation of the nonlinearity. The solid line shows the true nonlinearity.

6

An Identification Algorithm for Wiener Models

6.1 The Algorithm

We are now ready to state the Wiener model estimation algorithm.

Input: The data set $\{u(t), y(t)\}_{t=1}^{N}$, where $u(t)$ is the input of the system and $y(t)$ is the output.

Output: The parameter estimates $\hat{\theta}$ and $\hat{\eta}$, where $G(q, \hat{\theta})$ is the linear dynamic system and $f(., \hat{\eta})$ is the static nonlinearity.

Step 1: (Initial estimate, Section 4.1.) Parameterize the linear system with an FIR model,

$$x(t) = b_1 u(t-1) + \cdots + b_{n_b} u(t-n_b) \tag{6.1}$$

and the inverse of the nonlinear system as linear B-splines:

$$x(t) = f_1 B_1\big(y(t)\big) + \cdots + f_{n_f} B_{n_f}\big(y(t)\big) \tag{6.2}$$

Equating Equations (6.1) and (6.2), and setting $b_1 = 1$, this system of equations can be solved with linear regression to give estimates of b_i and f_i, as described in Section 4.1. Alternatively, the Instrumental Variable or Total Least squares method (Sections 6.3.2 and 6.3.3 respectively) may be used.

Step 2: (Model reduction, Chapter 5.) Use the FIR estimate to obtain an initial estimate of your desired model structure (e.g., output error), either

a) by using the FIR model to simulate $x(t)$, and estimating a linear model from the $\{u(t), x(t)\}$ data set, or

b) by using balanced truncation.

This gives an initial estimate of the parameters θ of the linear system.

Transform the splines representation of the inverse of the nonlinearity into a hinging hyperplanes representation of the nonlinearity (see Section 3.2.5). The number of breakpoints may be reduced either using the `newnot` algorithm (Section 5.2), or by visual inspection. We thus obtain an initial estimate of η, the parameters of the nonlinear system.

Step 3: Formulate the prediction error criterion described in Section 2.1:

$$V_N(\theta, \eta) = \frac{1}{N} \sum_{t=1}^{N} (y(t) - \hat{y}(t, \theta, \eta))^2 \tag{6.3}$$

Start with the estimate obtained in Step 2, and use Gauss-Newton minimization (see Section 2.4) to numerically find the values of θ and η that minimizes (6.3).

6.2 Consistency: Noise Free Case

Until now, we have mostly discussed the estimated Wiener model, and said very little about the underlying system. In a practical situation, we may not know much about the physical system; we try to describe it with a Wiener model, but we cannot expect the description to be exact. All we can do is to calculate the mean square prediction error or other error measures, and, depending on the application, decide if our model is good enough.

Now suppose that the physical system can be described exactly by a Wiener model, i.e., there exist "true" parameter values θ_0 and η_0, such that $x(t) = G(q, \theta_0)u(t)$ and $y(t) = f\big(x(t), \eta_0\big)$. Will the Wiener model estimation algorithm then give us the correct parameters? Another way to phrase this question: are the parameter estimates consistent? Even in the noise free case, this is not a trivial question. As remarked, the prediction error criterion may have, and will often have several local minima. The estimates obtained from one of these minima will in general not be consistent.

We will answer the consistency question in the form of a theorem.

Theorem 2 *Suppose that the true system is described by the following equation:*

$$y(t) = f\big(G(q,\theta_0)u(t), \eta_0\big) \tag{6.4}$$

Suppose also that the linear system G is stable, and that the nonlinear function $f(\cdot,\eta)$ is differentiable with a uniformly continuous first order derivative on $\mathbf{R}$ (the set of real numbers).

Further assume that

1. *The linear model structure is globally identifiable*
2. *The input data set is informative enough*
3. *The input to the nonlinearity, $\{x(t)\}_{t=1}^{N}$, is dense on $\mathbf{R}$ when N tends to infinity.*
4. *The number of parameters in the initial estimate, n_b and n_f, as well as the number of data, N, tends to infinity in such a way that*

$$\frac{n_b}{N} \to 0 \quad \text{and} \quad \frac{n_f}{N} \to 0 \tag{6.5}$$

The parameter estimates $\hat{\theta}$ and $\hat{\eta}$ obtained from the algorithm stated in Section 6.1 are then consistent. The consistency here excludes a constant gain that can be arbitrarily distributed between the linear and nonlinear subsystem.

If the derivative of f is uniformly continuous only on a subset of $\mathbf{R}$, the estimate of the nonlinearity will be consistent on that subset.

Proof

We will go through the algorithm step by step, and show that what is obtained in each step describes the system arbitrarily well.

Step 1: The first step consists of the initial estimate. A stable linear system can always be described arbitrarily well by an FIR model if the number of parameters is large enough, as mentioned in Section 3.1.2. (Note that this holds also for Laguerre and Kautz models.) Similarly, a differentiable function, with uniformly continuous first order derivative on a set, can be approximated arbitrarily well by a piecewise linear function with enough breakpoints (Proposition 3.1). Since the inverse of a continuously differentiable function (if it exists) is also

continuously differentiable, this will hold also for f^{-1}. Note that if f is piecewise continuously differentiable, the reasoning can be applied to each piece.

We thus know that there exist parameter values b_i^0 and f_i^0 that describe the linear system and the inverse nonlinearity arbitrarily well, and hence the error between $x(t) = G(q,\theta)u(t) = \sum b_i^0 u(t-i)$ and $x(t) = f^{-1}\big(y(t)\big) = \sum f_i^0 B_i\big(y(t)\big)$ will tend to zero for these values. Since the linear regression estimate is the one that minimizes this error, the estimates obtained in the first step will be exactly these values.

Step 2: If the FIR model of the linear system obtained in the first step arbitrarily well describes the linear system, a simulated output $\hat{x}$ will be arbitrarily close to the true intermediate signal $x(t)$ (apart from a constant gain). An output error model estimated from $u(t)$ and $\hat{x}(t)$ will then be consistent, as follows from consistency results for linear systems (Ljung, 1999).

Similar results hold for the balanced truncation (see Zhou et al., 1995). If the true system can be described by a state-space model with a smaller number of states, the redundant states will correspond to the smallest singular values, and can be removed without affecting the model.

The inverse of a piecewise linear function is also piecewise linear, so the inversion of the nonlinearity will not pose any problems. (Since the nonlinearity was assumed invertible, an accurate enough estimate will also be invertible).

If the true nonlinearity is piecewise linear with fewer breakpoints than used in the (consistent) estimate, some breakpoints will be redundant. These breakpoints may be removed without affecting the accuracy of the approximation.

Step 3: The noise free case is a special case of the general prediction error method consistency described in Section 2.2. The predictor is here exactly $\hat{y}(t,\theta,\eta) = f\big(G(q,\theta)u(t),\eta\big)$. The conditions stated in the theorem will assure that $\theta = \theta_0$ and $\eta = \eta_0$ in the minimum of $V_N(\theta,\eta)$. Since the previous two steps leads to consistent estimates, the Gauss-Newton minimization will lead to the global minimum.

□

6.3 Consistency with Noise: Initial Estimate

6.3.1 Linear Regression

The consistency question is more difficult when we have noise in the system. Recall from Section 2.4.2 that if

$$y(t) = \varphi(t)^T \Theta \tag{6.6}$$

the linear regression estimate is

$$\widehat{\Theta} = \left[\frac{1}{N} \sum_{t=1}^{N} \varphi(t)\varphi(t)^T \right]^{-1} \frac{1}{N} \sum_{t=1}^{N} \varphi(t) y(t) \tag{6.7}$$

If we assume that the true system is described by

$$y(t) = \varphi(t)^T \Theta_0 + w(t) \tag{6.8}$$

where $w(t)$ is noise, the linear regression estimate will be

$$\widehat{\Theta} = \Theta_0 + \left[\frac{1}{N} \sum_{t=1}^{N} \varphi(t)\varphi(t)^T \right]^{-1} \frac{1}{N} \sum_{t=1}^{N} \varphi(t) w(t) \tag{6.9}$$

Consistency results for linear regression estimates can be found in Ljung (1999). Under general conditions, we have that

$$\overline{E}\varphi(t)w(t) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \varphi(t)w(t) = h \tag{6.10}$$

The estimate will be consistent, i.e., $\widehat{\Theta} = \Theta_0$, if $h = 0$. This means that the noise, $w(t)$, has to be uncorrelated with the regressors, $\varphi(t)$.

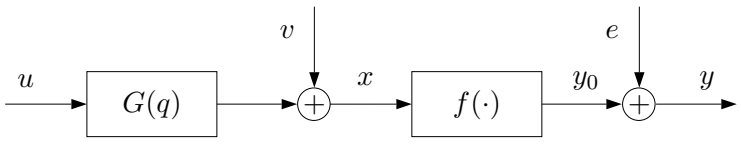


Figure 6.1: A Wiener system with noise. v denotes process noise and e measurement noise.

In Figure 6.1, a Wiener system with noise is depicted. As before, we assume that the system can be exactly described by an FIR model. We want to write this on the form (6.8). Fixing b_1 to 1, we get

$$-u(t-1) = \varphi(t)\Theta_0^T + w(t) \tag{6.11}$$

First assume that the measurement noise is zero. We then have

$$\varphi(t)^T = \big(u(t-2) \quad \dots \quad u(t-n_b) \quad -B_1\big(y(t)\big) \quad \dots \quad -B_{n_f}\big(y(t)\big)\big) \tag{6.12}$$

$$\Theta_0^T = \begin{pmatrix} b_2^0 & \dots & b_{n_b}^0 & f_1^0 & \dots & f_{n_f}^0 \end{pmatrix} \tag{6.13}$$

$$w(t) = v(t) \tag{6.14}$$

Since $y(t)$, and thus $B_i\big(y(t)\big)$, depends on the process noise $v(t)$, $w(t)$ will *not* be uncorrelated with the regressors. Hence, we cannot expect the initial estimate to be consistent if we have process noise.

Now instead assume that the process noise is zero, but that we have measurement noise. With the same regressor and parameter vector as above, $w(t)$ is then

$$w(t) = f^{-1}\big(y_0(t)\big) - f^{-1}\big(y(t)\big) \tag{6.15}$$

where $y(t) = y_0(t) + e(t)$ is the measured output. This is true since

$$f_1^0 B_1\big(y(t)\big) + \dots + f_{n_f}^0 B_{n_f}\big(y(t)\big) = f^{-1}\big(y(t)\big) \tag{6.16}$$

We will not have consistency in this case either, since this $w(t)$ (except in very special cases, like if f is linear) will be correlated with the regressors $\varphi(t)$.

6.3.2 Instrumental Variables

As described in Section 2.4.3 (again, see also Ljung, 1999), the IV method may yield a consistent estimate in cases where linear regression fail to do so. Recall that the IV estimate is (in the case of a Wiener model) the solution of

$$\frac{1}{N}\sum_{t=1}^{N} \zeta(t)\big(-u(t-1) - \varphi(t)^T\Theta\big) \tag{6.17}$$

where $\varphi(t)$ is the regressor from Equation (6.12). The solution exists uniquely if the matrix $\frac{1}{N}\sum \zeta(t)\varphi(t)^T$ has full rank $(n_b + n_f - 1)$. When N tends

to infinity, this means that the instruments should be correlated with the regressor.

Let, as before, the true system be described by Equation (6.11). The IV estimate will be consistent, when the number of data tends to infinity, if

$$\lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} \zeta(t) w(t) = 0 \tag{6.18}$$

Now, how do we select instruments that are correlated with the regressor but not with the noise $w(t)$? Looking first at the case with *only process noise*, we have

$$w(t) = v(t) \tag{6.19}$$

The input $u(t)$ is a deterministic signal and thus uncorrelated with the process noise, but the output $y(t)$ will depend on the noise. To get around this, we first make a linear regression estimate of the system. This estimate can then be used to simulate noise free output, $\hat{y}(t)$. We select the instruments as

$$\zeta(t)^T = \begin{pmatrix} u(t-2) & \dots & u(t-n_b) & -B_1\big(\hat{y}(t)\big) & \dots & -B_{n_f}\big(\hat{y}(t)\big)\big) \end{pmatrix} \tag{6.20}$$

These instruments will be correlated with the regressor since the data comes from the same input $u(t)$, but it will be uncorrelated with the noise since $\hat{y}(t)$ comes from a noise free simulation.

In the case when we instead have *only measurement noise*, we have

$$w(t) = f^{-1}\big(y_0(t)\big) - f^{-1}\big(y(t)\big) \tag{6.21}$$

Except for very special cases (like f being linear), it is hard to generate instruments that are correlated with the regressors (which depend on $y(t)$), but that are uncorrelated with $w(t)$. A few sufficient, but very restrictive conditions when this holds are detailed in Appendix A.

6.3.3 Total Least Squares

The Total Least Squares, TLS, method is described in Van Huffel and Vanderwalle (1991), and was mentioned in Chapter 4. The TLS method finds the solution to the problem

$$\min_{\Theta\in\mathbf{R}^{\mathbf{n}}} \|\Phi\Theta\| \quad \text{subject to} \quad \Theta^T\Theta = 1 \tag{6.22}$$

where Φ is the data matrix,

$$\Phi = \begin{pmatrix} \varphi(1)^T \\ \dots \\ \varphi(N)^T \end{pmatrix} \tag{6.23}$$

The TLS solution is unique if $\sigma_{n-1} > \sigma_n$, where σ_i are the singular values of the data matrix, with σ_1 the largest and σ_n the smallest such value. When we have no noise, this only differs with a scaling factor from the linear regression estimate.

The TLS estimate is consistent in the case of "errors-in-variables". Assume that the data is related as

$$\Phi_0 \Theta_0 = 0 \tag{6.24}$$

but what we observe is $\Phi = \Phi_0 + \Delta\Phi$. $\Delta\Phi$ are the measurement errors. Further assume that each row of $\Delta\Phi$ is independent and identically distributed with zero mean and covariance $\sigma_v I$. This means that the errors are uncorrelated, and have the same variance. The TLS in this case, unlike the regular linear regression estimate, gives a consistent estimate of Θ_0.

The Wiener model, unfortunately, does not fit into the errors-in-variables model. We have

$$\varphi(t)^T = \begin{pmatrix} u(t-1) & \dots & u(t-n_b) & -B_1\big(y(t)\big) & \dots & -B_{n_f}\big(y(t)\big)\big) \end{pmatrix} \tag{6.25}$$

where the input $u(t)$ is known exactly, while the output $y(t)$ is measured with noise. Since the basis functions B_i all depend on $y(t)$, the errors will be correlated. Furthermore, the Equation (6.24) will not hold if we have process noise. The TLS estimate will thus *not* be consistent.

The special structure of the Wiener model may cause additional problems when using the TLS method. Since the support for the B-splines basis functions is small, the data matrix Φ will be sparse when using many basis functions. The solution of the TLS problem (see Van Huffel and Vanderwalle, 1991) is the right singular vector corresponding to the smallest singular value of Φ. If one basis function lacks support from data, the corresponding singular value may be unreasonably small. Van Huffel and Vanderwalle (1991) also claims that TLS is less robust than linear regression.

6.3.4 Conclusions

In this section we have studied the initial estimate of the Wiener model, when the true system contains noise. We summarize the general results – there are exceptions for very special nonlinearities, like f being linear.

- The linear regression estimate is not consistent, neither with measurement noise nor process noise.

- If we have only process noise, the linear regression estimate can be used to simulate noise free output data. Instruments constructed from these noise free data will then give a consistent IV estimate of the Wiener system.

- If there is measurement noise, the IV estimate will not be consistent.

- The TLS estimate is in general not consistent for the Wiener model. The method is also less robust than the linear regression method.

6.4 Consistency with Noise: Prediction Error Estimate

The consistency of the prediction error estimate was discussed in Section 2.2. Under conditions on identifiability of the linear model, information content in the input data, and that the nonlinearity was invertible, it was shown that the prediction error estimate is consistent. The predictor could sometimes be hard to formulate, but under additional noise assumptions, a simplified predictor could be used.

The prediction error criterion cannot be minimized analytically, but numerical search methods has to be used. Furthermore, the criterion may have several local minima, so a good initial estimate is needed for the numerical search. **To find the prediction error estimate is hard also in the noise free case.**

If the initial estimate is consistent, and we use the true predictor, the Gauss-Newton minimization will lead to the global minimum of the prediction error criterion, and consistency will be assured. But this is not a necessary condition; it suffices that the initial estimate is in the attraction region of the global minimum.

In the noise free case, the initial estimate via linear regression, presented in Chapter 4, will describe the true system arbitrarily well. The true predictor is simple to formulate, so the numerical minimization will lead to the global minimum, and the prediction error estimate will be consistent under the conditions mentioned above.

With only measurement noise and no process noise in the system, the initial estimate is not consistent. However, the prediction error estimate, as

shown in Section 2.2, will be consistent. For small noise levels (large signal-to-noise ratios), the initial estimate will be close to the true system, and the numerical minimization will lead to the global minimum, thus giving a consistent estimate.

If the system contains process noise but no measurement noise, it was shown in the previous section that an IV estimate will give a consistent initial estimate. The true predictor may be hard to express. In Section 2.2 it was shown that under additional conditions on the noise and the nonlinearity, a simplified predictor may be used and will yield a consistent estimate. The true predictor can also be approximated with an extended Kalman filter. With the consistent initial estimate, the minimum obtained for the approximative predictor will be close to the true (consistent) minimum.

To quantify the noise levels that will give a consistent estimate even if the initial estimate is not consistent is hard, since the attraction region of the global minimum is unknown. It is also very hard to analyze the properties of the model reduction and the prediction error criterion minimization in detail, and to quantify the errors they introduce if the initial estimate is erroneous.

On the other hand, even if the initial estimate cannot be proved to be consistent, it may still be good enough. Since the noise free estimate is consistent, a small noise perturbation will (from continuity reasons) only move the estimate a small distance from the noise free optimum. This means that the numeric search algorithm may still lead us to the global minimum of the prediction error criterion.

Using wavelets or radial basis neural networks, we may obtain a consistent estimate of any nonlinear system if we use enough basis functions (Delyon et al., 1995). If we then want to convert it to the Wiener structure, the algorithm presented in this chapter may be useful. The black-box model may be used to simulate data. These data will be noise free and hence allow for a consistent estimate.

This is in theory a very attractive method, since it gives a consistent estimate. In practice, the number of basis functions needed is often unreasonably large (Sjöberg et al., 1995). This will put high demands both on the number of data and on the computational power.

7

Examples

In this chapter we look at some examples using the Wiener model structure. We show how models within this class can be identified using the methods proposed in this thesis. Using Monte Carlo simulations, we show that although questions still remain about the consistency of the estimates (see Chapter 6), the initial estimate may lead to a good final estimate. Most of the examples are from simulated data, which has the advantage that we can compare the estimate with the true system, and also that we can easily increase the number of data. We also have one example with real data from a distillation column.

We have tried to compare our method with other methods used. This is not always easy to do in a "fair" way. For example, if we use a saturation function that is piecewise linear, and can be described exactly using hinging hyperplanes, is it then fair to compare it with a method that uses Chebyshev polynomials? We have evaluated our estimates using the prediction error criterion. For a method described in an article using another measure, where the code is not available, this is hard to compute. Even when we have the code, the comparison is not straight-forward. There are always design parameters to choose, as the order of the linear system, and the number of parameters of the nonlinearity. Using the same number of parameters need not be equivalent between two different model structures.

The algorithm is also compared with the alternative initialization mentioned in the motivating example in Section 1.3. We assume that the data

comes from a linear system and we start with estimating an output error model from input-output data. We use the output error model to simulate the intermediate signal $x(t)$, and plot it against the measured output $y(t)$. Thereafter we estimate the nonlinearity from the simulated $x(t)$ and the measured $y(t)$ with linear B-splines. This estimate is converted to hinging hyperplanes, and the prediction error criterion is minimized with a Gauss-Newton search.

When estimating models from noisy data, the particular noise realization will affect the estimate. The parameters may adapt not only to reflect the system properties, but also to the noise. This is an undesired effect, since the next time we collect data from the system the noise realization will be different, but we still want our model to represent the same system. To avoid that this over-adaptation is reflected in the performance measure we use, the prediction error criterion $V_N(\theta, \eta)$, the criterion may be evaluated on a "fresh" data set. This is a data set that is not used for estimation. It is denoted *validation data*.

Some of the following examples are "standard" examples from articles in this field (Wigren, 1993; Kalafatis et al., 1997; Bruls et al., 1997)). We have then tried to use the same input to be able to compare with the results obtained in those articles.

7.1 Motivating Example

We first return once again to the example from the introduction. Recall that the system was described by

$$x(t) = \frac{q^{-1}}{1 - 1.40q^{-1} + 0.49q^{-2}}u(t) \tag{7.1}$$

$$y(t) = e^{x(t)} \tag{7.2}$$

and the input signal was a sum of sinusoids:

$$u(t) = \sum_{k=1}^{20} \sin(k\pi t/10 + \phi_k) \tag{7.3}$$

where ϕ_k is a stochastic variable with uniform distribution on $[0, 2\pi]$.

Starting with 20 FIR parameters and 10 linear B-splines, fixing b_1 to 1 and spreading the breakpoints with even support from data (all as described in Section 4.2.1), we obtained an initial estimate of the linear and the nonlinear system.

The estimate of the linear system was used to simulate the intermediate signal $x(t)$. The original input $u(t)$ was used for this. An output error model of the same order as the true system was then estimated from the $(u(t), x(t))$ data. The following estimate was obtained:

$$G(q, \theta^{(1)}) = \frac{0.97q^{-1}}{1 - 1.42q^{-1} + 0.51q^{-2}} \tag{7.4}$$

The estimate of the nonlinearity is visualized by plotting the simulated $x(t)$ versus the measured $y(t)$ in Figure 7.1. The number of breakpoints was then reduced to 5 using the `newnot` algorithm. This was then converted to hinging hyperplanes parameters.

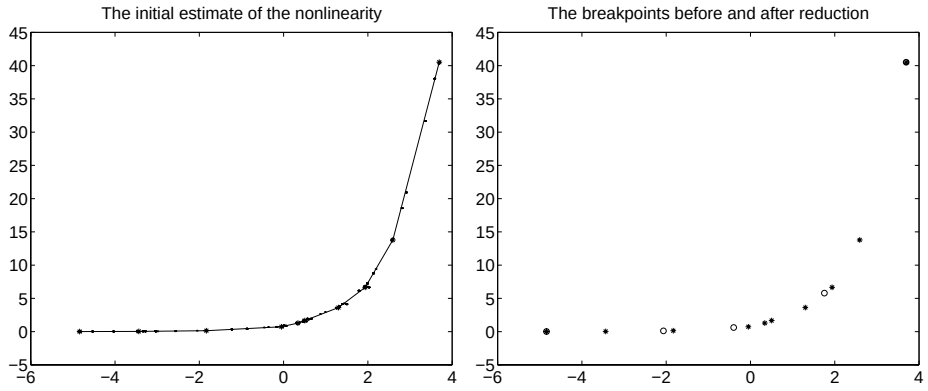


Figure 7.1: The initial estimate of the nonlinearity. The breakpoints are marked with stars, the ones kept after the reduction with circles.

After the numerical minimization the following estimate was obtained:

$$G(q, \hat{\theta}) = \frac{0.97q^{-1}}{1 - 1.41q^{-1} + 0.50q^{-2}} \tag{7.5}$$

The estimate of the nonlinearity is plotted, together with the simulated $x(t)$ and measured $y(t)$, to the left in Figure 7.2. The Gauss-Newton search needed 9 iterations. The minimal value of the prediction error criterion was

$$V_N(\hat{\theta}, \hat{\eta}) = 0.2321 \tag{7.6}$$

An alternative way of initializing the parameters is to estimate a linear output error model directly from input-output data. The nonlinearity may

then be estimated from simulated $x(t)$ and measured $y(t)$. This initial estimate led to a minimum of the prediction error criterion at

$$G(q, \hat{\theta}) = \frac{3.23q^{-1}}{1 - 1.41q^{-1} + 0.50q^{-2}} \tag{7.7}$$

The minimum value of the criterion was 0.2321, and 10 Gauss-Newton iterations were needed. The nonlinearity is plotted to the right in Figure 7.2.

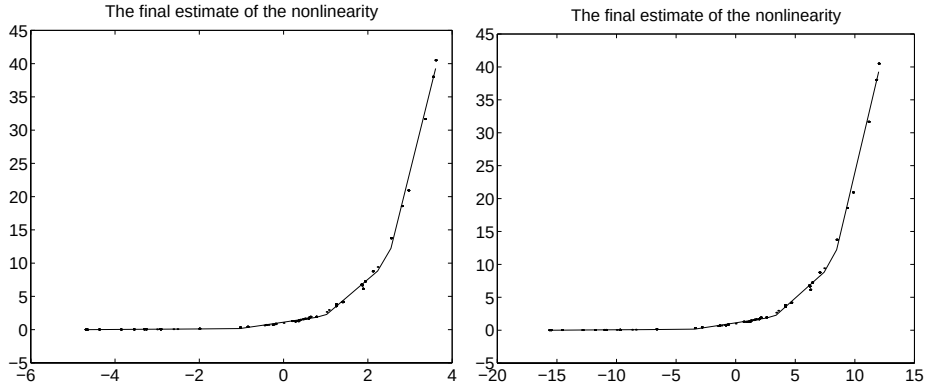


Figure 7.2: Final estimate of the nonlinearity. Simulated x is plotted against measured y. The solid line shows the estimated nonlinearity. Left: Using linear regression initialization. Right: Using output error initialization.

We find that both estimates are very close to the true system. To further improve the accuracy, more breakpoints are needed for the nonlinear system.

In this case, there is thus no difference between initializing the numerical search with the linear regression method, or estimating a linear model from input-output data. The linear regression estimate started slightly closer to the minimum, and needed one Gauss-Newton step less than the linear model initialization.

7.1.1 Using Noisy Data

Real life data are always measured with some noise. In this section the same system is used, but we add artificial measurement noise.

$$y(t) = e^{x(t)} + e(t) \tag{7.8}$$

where $e(t)$ is white Gaussian noise, with variance $\sigma^2 = 1$.

Since different noise realizations may affect the estimation, we have used a Monte Carlo simulation, with 500 independent data sets. The prediction error criterion was minimized using a Gauss-Newton numerical search. The model obtained when the numerical search was initialized with the linear regression was compared with the model obtained when an output error model estimated from input-output data of the system was used for initialization. Validation data was generated the same way as the estimation data, and the prediction error criterion was calculated for this validation data set.

In Figure 7.3, the criterion values obtained for the linear regression initialization are plotted against the criterion values obtained from the output error initialization. A large number of points are close to the $(1, 1)$ corner. This is the optimal value, since the measurement noise has variance 1. One may note that in most cases the two initialization methods lead to different minima. In roughly half these cases, the linear regression leads to a smaller value, in the other cases to a larger. We may thus not conclude that one method is preferable in general; both has their advantages. This will depend on the particular noise realization, so we cannot say in advance which one is better for a given data set.

7.2 A Non-Invertible Nonlinearity

The motivation and proofs for the suggested algorithm rely heavily on the assumption that the nonlinearity is invertible. Here we try the algorithm on a system where the nonlinearity is not invertible. Even though the initial estimate will not be consistent, it can still be useful and may lead to a good final estimate.

The example is the same as in Section 4.2.2.

$$x(t) = \frac{q^{-1}}{1 - 0.7q^{-1}} u(t) \tag{7.9}$$

$$f(x) = \begin{cases} -0.1x - 1.1 & \text{if } x < -1 \\ x & \text{if } -1 \leq x < 1 \\ -0.1x + 1.1 & \text{if } 1 \leq x \end{cases} \tag{7.10}$$

The input $u(t)$ was white Gaussian noise with variance 1. No measurement noise was added.

The true nonlinearity is depicted to the left in Figure 7.4. The initial estimate shown to the right in the figure was calculated using 20 FIR and

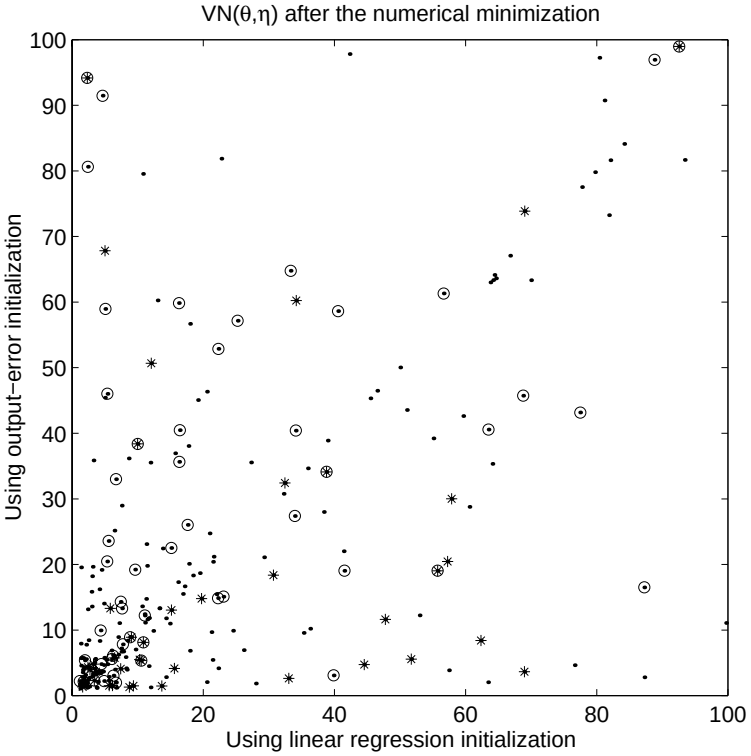


Figure 7.3: Value of the prediction error criterion $V_N(\theta, \eta)$ for validation data after minimization using different initialization algorithms. The value obtained using linear regression initialization is plotted on the x-axis, against the value obtained using an output error estimate. If the search algorithm did not reach a minimum after 50 iterations, the value is marked by a star (if it was for the linear regression initialization) and/or a circle (if it was for the output error initialization).

10 B-splines parameters. The simulated $x(t)$ versus the measured $y(t)$ is plotted to the right in Figure 7.4. The data points are somewhat scattered, but the shape of the true nonlinearity can clearly be seen (though the scale on the x-axis is slightly different). We may thus keep the initial estimate of the linear subsystem, and use the simulated data to make a better estimate of the nonlinearity.

We decide to use four breakpoints, since the plot suggests that the nonlinearity is piecewise linear and consists of three pieces. We select the breakpoints to the minimal and maximal value of the simulated x, and to -1 and

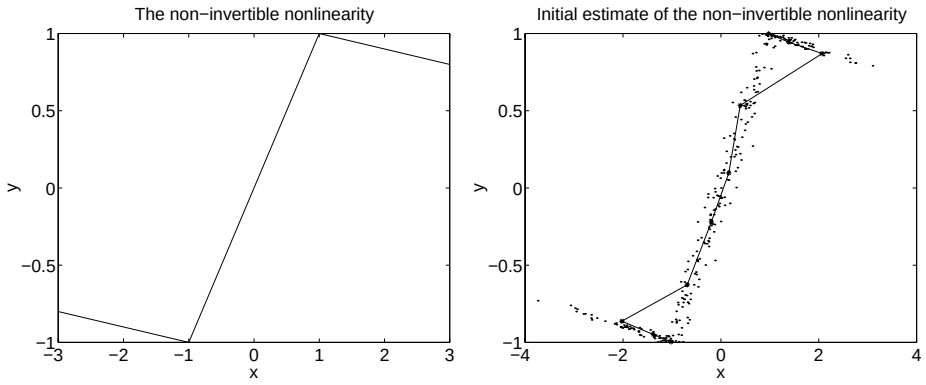


Figure 7.4: Left: The non-invertible saturation-like nonlinearity. Right: The linear regression estimate of the nonlinearity. Simulated x is plotted against measured y. The solid line shows the estimated nonlinearity.

1. We then estimate the corresponding B-spline parameters. The estimate is converted to hinging hyperplanes, and a Gauss-Newton search is initialized with these values. The linear part is initialized with an output error model estimated from u and the simulated $x(t)$.

The final estimate is shown in Figure 7.5. The estimate of the linear subsystem was

$$G(q) = \frac{0.96q^{-1}}{1 - 0.70q^{-1}} \tag{7.11}$$

Apart from a scaling factor, it is very close to the true system. The value of the prediction error criterion was $V_N(\theta, \eta) = 2.1 \cdot 10^{-32}$. This example shows that the linear regression initial estimate may be useful also in cases when it is not consistent. With small adjustments, it led in this case to the global minimum.

7.3 A Control Valve Model

This example is taken from Wigren (1993) and describes a valve for control of fluid flow. The example is also used in Kalafatis et al. (1997). $u(t)$ is here the pneumatic control signal and $x(t)$ is the position of the valve plug. $y(t)$ is the resulting flow. The relations between u, x and y are described by

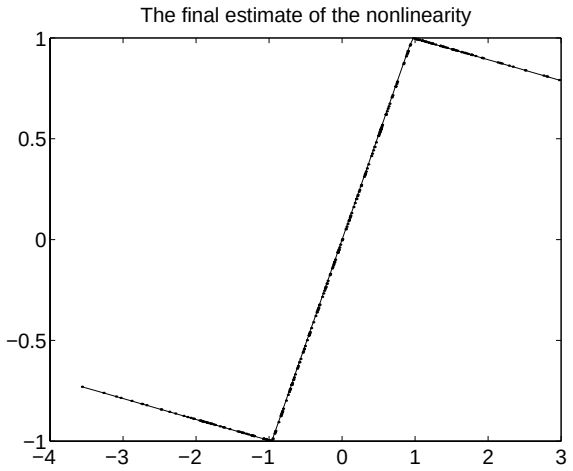


Figure 7.5: The final estimate of the non-invertible nonlinearity. Simulated x is plotted against measured y. The solid line shows the estimated nonlinearity.

the following equations:

$$\begin{aligned} x(t) = G(q)u(t) &= \frac{0.1044q^{-1} + 0.0883q^{-2}}{1 - 1.4138q^{-1} + 0.6065q^{-2}}u(t) \\ y(t) = f(x(t)) &= \frac{x(t)}{\sqrt{0.10 + 0.90(x(t))^2}} \end{aligned} \tag{7.12}$$

The input signal was generated using a PRBS with a basic clock period of seven sampling intervals, switching between -1 and 1. In each time interval of constant signal level, the signal was multiplied with a random factor uniformly distributed between 0 and 0.4, and a bias of 0.5 was added. This is the same procedure as the one used in Wigren (1993), and gives a signal with amplitude between 0.1 and 0.9. White Gaussian measurement noise with standard deviation 0.05 was added to the output. The data is shown in Figure 7.6.

We applied our estimation algorithm to these data. The number of FIR parameters was chosen to 30, the initial number of linear B-splines was 10. The intermediate signal $x(t)$ was then simulated using the FIR model, and a second order output error model was estimated from the simulated data. (This is the same model order as the true system). The number of breakpoints was reduced to 5 using the `newnot` algorithm.

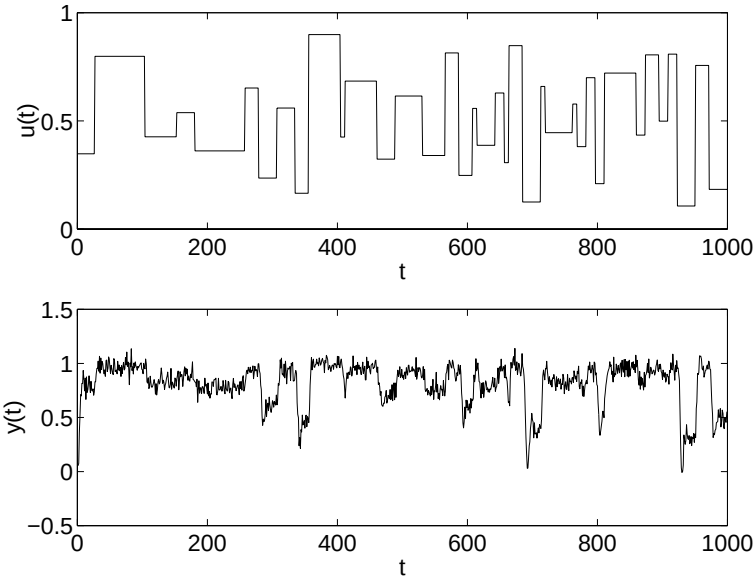


Figure 7.6: Input and output signal for the valve example.

After the numerical minimization, the following linear model was obtained:

$$G(q) = \frac{0.1057q^{-1} + 0.1014q^{-2}}{1 - 1.4204q^{-1} + 0.6134q^{-2}} \tag{7.13}$$

The estimated nonlinearity is shown in Figure 7.7. The value of the prediction error criterion was 0.0024.

According to the plots in Wigren (1993) and Kalafatis et al. (1997), they have obtained similar results. They both used prior knowledge to position the breakpoints. An exact comparison is hard to make, but we may note some details from our estimation procedure.

- The initial estimate will strongly depend on how the gain is fixed. In this case, both fixing one of the FIR parameters to one, and fixing one of the B-splines parameters gave very bad initial estimates. Fixing the steady-state gain of the linear system, i.e., fixing the sum of the FIR parameters to one, gave a much better estimate of both the linear and the nonlinear subsystem.

- Even if the linear system is accurately described by the initial FIR estimate, the estimate of the nonlinearity may be affected by the noise,

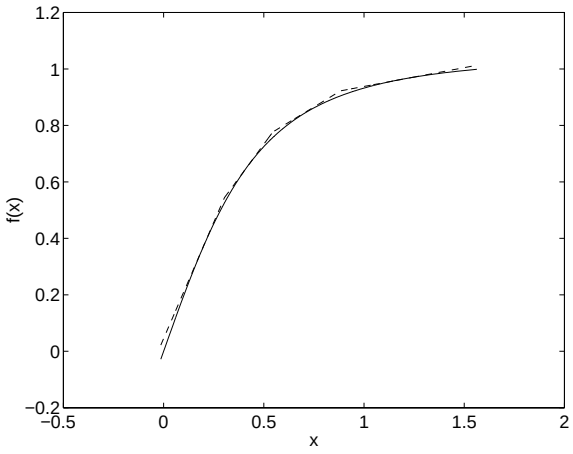


Figure 7.7: The estimate of the nonlinearity in the valve model. The solid line shows the true nonlinearity, the dashed line the estimated one.

and sometimes not even invertible. This is easily discovered on a plot of simulated x versus measured y, and the estimated nonlinearity. If the estimate does not seem reasonable, a better one may be obtained by estimating B-splines directly from the simulated x and the measured y, as we did in the example in Section 7.2. (This was however not the case for the data set shown in Figure 7.6.)

7.4 A Distillation Column

As a real-life example, we have used data from a distillation column. These are the same data as in Bruls et al. (1997). A plot of the data is shown in Figure 7.8. The input is sampled every 2 minutes. The output is sampled every 18 or 20 minutes, and is assumed to be constant between the sampling times.

The authors of Bruls et al. (1997) argue that the data contains a time shift of 28 minutes or 14 samples. The data was thus shifted to account for this. Since there were relatively few data, all data were used for estimation. The estimated model was then validated on the same data. To select the number of FIR and spline parameters was not a trivial task, but after some trials, we settled for 400 FIR parameters and 10 spline parameters. The number of FIR parameters may seem very large, but it turns out that the

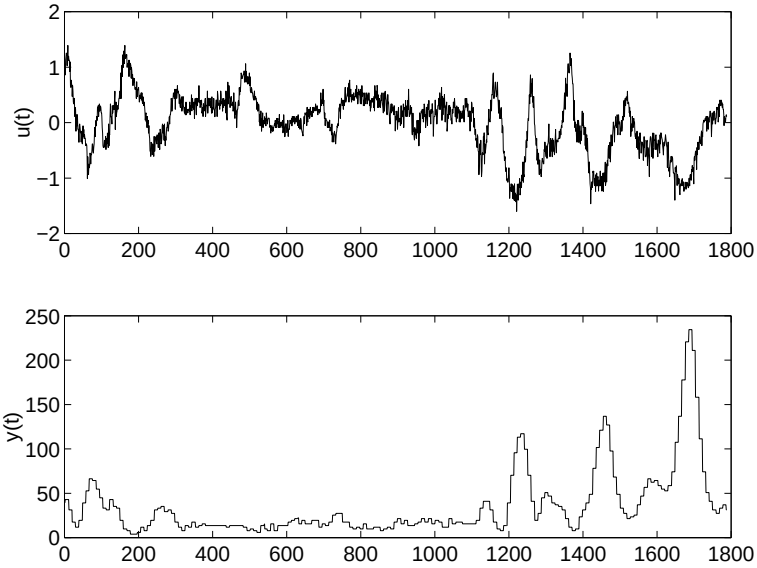


Figure 7.8: Data from the distillation column. Top: input, bottom: output.

linear subsystem has a very slowly decaying impulse response. The first nonlinear parameter was fixed to -1. The model obtained in Bruls et al. (1997) was of order 2, so the initial estimate was reduced to a second order OE model. The number of breakpoints were reduced to 5, to yield a reasonable comparison with Bruls et al. (1997), who uses fifth order Chebyshev polynomials.

The value of the prediction error criterion for the final estimate was 44.5. The estimated and measured outputs are shown in Figure 7.9. In Figure 7.10 the estimated x is plotted versus the measured y. The estimated nonlinearity is also plotted.

We may compare these results with those obtained with the separable least squares method proposed in Bruls et al. (1997). Using the same data and estimating a second order state space model for the linear subsystem and fifth order Chebyshev polynomials, a mean square fit of 52.2 was obtained. The state space model was estimated using the SMI toolbox (Haverkamp and Verhaegen, 1997). Figures 7.11 and 7.12 show the estimates.

To draw any conclusions about which method is preferable from this small data sample is of course hazardous. It is still reasonable to say that

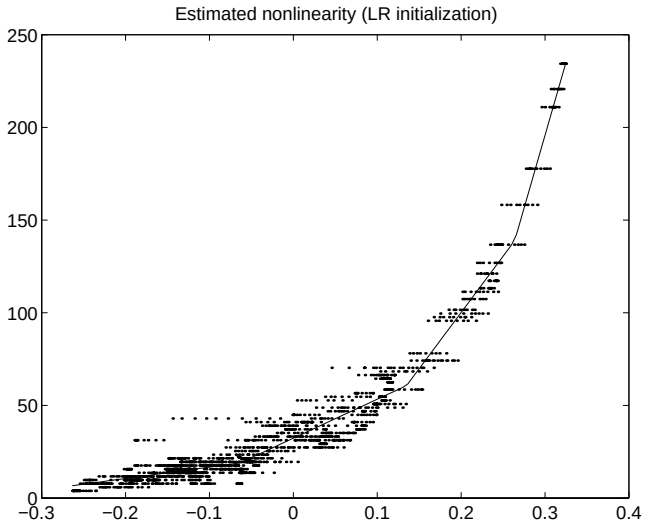


Figure 7.9: Simulated and measured output of the distillation column, using the linear regression initialization. The solid line is the measured output, the dots show the simulated output. The nonlinearity is modeled as a piecewise linear function.

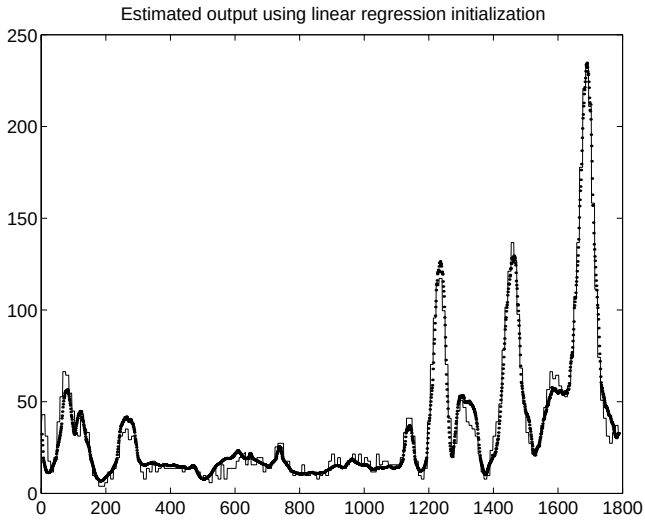


Figure 7.10: The estimated nonlinearity of the distillation column. Simulated x is plotted against measured y. The solid line shows the estimated nonlinearity.

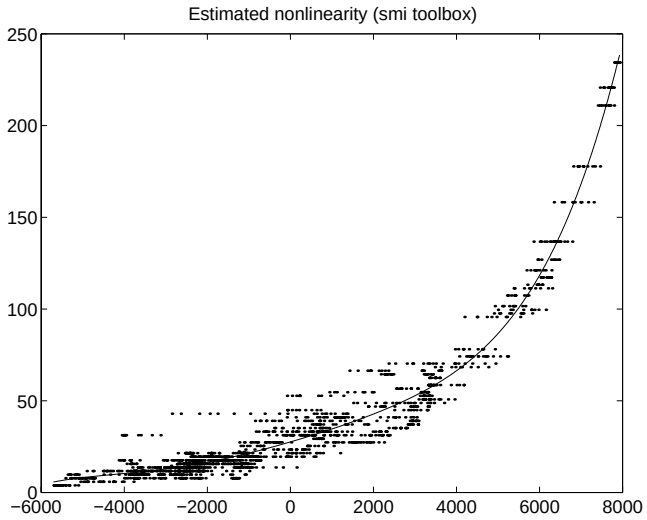


Figure 7.11: Simulated and measured output of the distillation column, using the separable least squares method. The solid line is the measured output, the dots show the simulated output. The nonlinearity is modeled with Chebyshev polynomials.

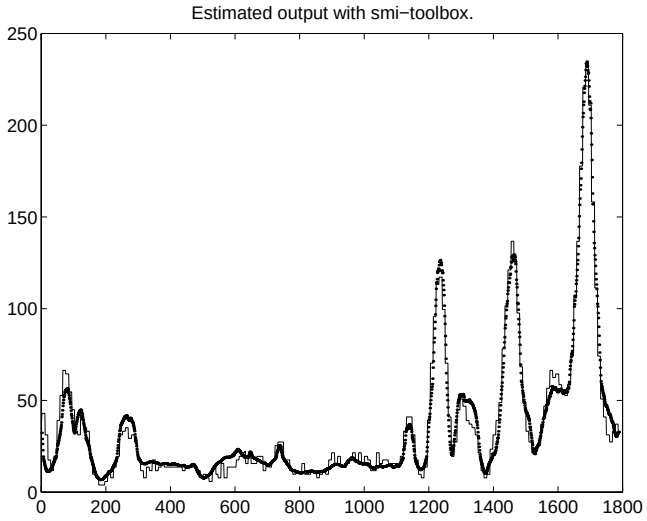


Figure 7.12: The estimated nonlinearity, using the separable least squares method. Simulated x is plotted against measured y. The solid line shows the estimated nonlinearity.

both methods are useful. It is also interesting to look at combinations of them. The linear regression estimate could very well be converted to a state space model, and the minimization continued with separable least squares. This is a subject for further research.

With the slowly decaying impulse response, it would also be interesting to try Laguerre models instead of FIR models for the initial estimate.

A

Appendix

A.1 IV with Measurement Noise

As seen in Section 6.3.2, the IV method may give a consistent initial estimate of the system if there is only process noise. This appendix details some conditions which will give consistent estimates also with measurement noise.

We suppose the true system is described by the following equation:

$$y_0(t) = f(G(q, \theta_0)u(t), \eta_0) \tag{A.1}$$

We assume that u can be measured without noise, but we cannot measure y, only

$$y(t) = y_0(t) + e(t) \tag{A.2}$$

The following relation describe the true system:

$$\sum_{i=1}^{n_b} b_i^0 u(t-i) = \sum_{i=1}^{M_b} f_i^0 B_i(y_0(t)) \tag{A.3}$$

We may multiply both sides with an instrument vector $\zeta(t)$ and sum over the time index t. N is the number of data.

$$\sum_{i=1}^{n_b} b_i^0 \frac{1}{N} \sum_{t=1}^{N} \zeta(t)u(t-i) = \sum_{i=1}^{M_b} f_i^0 \frac{1}{N} \sum_{t=1}^{N} \zeta(t)B_i(y_0(t)) \tag{A.4}$$

The IV estimate satisfies

$$\frac{1}{N}\sum_{t=1}^{N}\zeta(t)\left(\sum_{i=1}^{n_b}\hat{b}_i u(t-i)-\sum_{i=1}^{M_b}\hat{f}_i B_i(y(t))\right)=0 \tag{A.5}$$

or

$$\sum_{i=1}^{n_b}\hat{b}_i\frac{1}{N}\sum_{t=1}^{N}\zeta(t)u(t-i)=\sum_{i=1}^{M_b}\hat{f}_i\frac{1}{N}\sum_{t=1}^{N}\zeta(t)B_i(y(t)) \tag{A.6}$$

Note that we use the measured value $y(t)$ instead of the true $y_0(t)$ in the IV estimate. A unique solution can be obtained, e.g., by fixing one of the parameters. We assume in the following that we have a unique solution.

Comparing Equations (A.4) and (A.6) we see that the estimate will be consistent if

$$\frac{1}{N}\sum_{t=1}^{N}\zeta(t)u(t-i)=\frac{1}{N}\sum_{t=1}^{N}\zeta(t)u(t-i)\quad i=1,\ldots,n_b \tag{A.7}$$

$$\frac{1}{N}\sum_{t=1}^{N}\zeta(t)B_i\big(y_0(t)\big)=\frac{1}{N}\sum_{t=1}^{N}\zeta(t)B_i\big(y(t)\big)\quad i=1,\ldots,M_b \tag{A.8}$$

The first one is trivially true, but the second one poses more problems. It is clear that the consistency depend highly on the choice of basis functions. As said in Section 6.2 it is hard to say something about general consistency, but we will examine Equation (A.8) for a particular choice of basis functions: The linear B-splines described in Section 3.2.3.

The elements of Equation (A.8) have two different typical forms:

$$\frac{1}{N}\sum_{t=1}^{N}u(t-j)B_i\big(y_0(t)\big)=\frac{1}{N}\sum_{t=1}^{N}u(t-j)B_i\big(y(t)\big) \tag{A.9}$$

$$\frac{1}{N}\sum_{t=1}^{N}B_j\big(\hat{y}(t)\big)B_i\big(y_0(t)\big)=\frac{1}{N}\sum_{t=1}^{N}B_j\big(\hat{y}(t)\big)B_i\big(y(t)\big) \tag{A.10}$$

where $\hat{y}(t)$ is the noise free estimate of y from the initial linear regression model. j goes from 1 to n_b in the upper equation and from 1 to M_b in the lower one, i goes from 1 to M_b in both equations. Note that everything except $y_0(t)$ is known, so it can be computed numerically from data and estimates.

Recall the equations for the linear B-splines, where y_i are the fixed breakpoints.

$$B_i(y) = \begin{cases} 0 & \text{if } y < y_{i-1} \text{ or } y_{i+1} \leq y \\ \frac{y-y_{i-1}}{y_i-y_{i-1}} & \text{if } y_{i-1} \leq y < y_i \\ \frac{y_{i+1}-y}{y_{i+1}-y_i} & \text{if } y_i \leq y < y_{i+1} \end{cases} \tag{A.11}$$

Using Equation (A.11) we can expand the left side of Equation (A.9)

$$\frac{1}{N}\sum_{t=1}^{N} u(t-j)B_i(y_0(t)) = \frac{1}{N}\sum_{\substack{t \\ y_{i-1}\leq y_0(t)<y_i}} u(t-j)\frac{y_0(t)-y_{i-1}}{y_i-y_{i-1}} + \frac{1}{N}\sum_{\substack{t \\ y_i\leq y_0(t)<y_{i+1}}} u(t-j)\frac{y_{i+1}-y_0(t)}{y_{i+1}-y_i} \tag{A.12}$$

We do not know $y_0(t)$, since we can only measure $y(t)$, but we will now use that $y_0(t) = y(t) - e(t)$. Assume that e(t) is white Gaussian noise with zero mean and a known variance σ^2. Then for a given $y(t)$, $y_0(t)$ will be Gaussian with variance σ^2 and mean $y(t)$, and the probability that $y_{i-1} \leq y_0(t) < y_i$ can be computed as

$$P\big(y_{i-1} \leq y_0(t) < y_i | y(t)\big) = \int_{y_{i-1}}^{y_i} \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(\xi - y(t))^2}{2\sigma^2}\right\} d\xi \tag{A.13}$$

Instead of summing over the t:s where the unknown $y_0(t)$ lies in a certain interval, we can, as N tends to infinity, sum over all t, weighing the terms with the probability that $y_0(t)$, given $y(t)$, which is known, lies in that interval.

This does not quite solve the problem of computing (A.12) since the expression contains $y_0(t)$, but inserting that $y_0(t) = y(t) - e(t)$ allows us to split the first sum in two parts:

$$\frac{1}{N}\sum_{\substack{t \\ y_{i-1}\leq y_0(t)<y_i}} u(t-j)\frac{y_0(t)-y_{i-1}}{y_i-y_{i-1}} = \frac{1}{N}\sum_{\substack{t \\ y_{i-1}\leq y_0(t)<y_i}} \left\{u(t-j)\frac{y(t)-y_{i-1}}{y_i-y_{i-1}} - u(t-j)\frac{e(t)}{y_i-y_{i-1}}\right\} \tag{A.14}$$

The first term can now be computed as described above as a weighted sum over all t. The second term contains the unknown measurement noise $e(t)$.

To proceed we make the following rather strong assumption:

$$\frac{1}{N} \sum_{\substack{t \\ y_{i-1} \le y_0(t) < y_i}} u(t-j)e(t) \to 0 \text{ when } N \to \infty \tag{A.15}$$

Using this assumption, and the weighted summation described above we now have that the left hand side of Equation (A.9) can be computed numerically as

$$\begin{aligned} \frac{1}{N}\sum_{t=1}^{N} u(t-j)B_i(y_0(t)) = \\ \frac{1}{N}\sum_{t=1}^{N}\Big\{ u(t-j)\frac{y(t)-y_{i-1}}{y_i - y_{i-1}} P\big(y_{i-1} \le y_0(t) < y_i | y(t)\big) \\ + u(t-j)\frac{y_{i+1}-y(t)}{y_{i+1}-y_i} P\big(y_i \le y_0(t) < y_{i+1} | y(t)\big)\Big\} \end{aligned} \tag{A.16}$$

Although rather complicated and difficult to analyze, this expression is computable for given input-output data u and y, fixed breakpoints y_i of the linear splines, and known noise variance σ^2. Also the right hand side of the consistency constraint (A.9) is computable, and similar expressions can be derived for (A.10). It is thus possible to check during the identification if the estimates will be consistent. Note however that the assumption in Equation (A.15) is very strong, and it is not clear how to interprete it.

Bibliography

Al-Saggaf, U. M. and Franklin, G. F. (1988). Model reduction via balanced realizations: An extension and frequency weighting techniques. *IEEE Transactions on Automatic Control*, 33(7):687–692.

Anderson, B. D. O. and Moore, J. B. (1979). *Optimal Filtering*. Prentice-Hall, Englewood Cliffs, New Jersey, USA.

Baum, L., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41:164–171.

Bergman, N. (1998). Expectation maximization segmentation. Technical Report LiTH-ISY-R-2067, Department of Electrical Engineering, Linköping University.

Billings, S. A. and Fakhouri, S. Y. (1977). Identification of nonlinear systems using the Wiener model. *Electronics Letters*, 13(17):502–504.

Billings, S. A. and Fakhouri, S. Y. (1982). Identification of systems containing linear dynamics and static nonlinear elements. *Automatica*, 18(1):15–26.

Boyd, S. and Chua, L. O. (1985). Fading memory and the problem of ap-

proximating nonlinear operators with Volterra series. *IEEE Transactions on Circuits and Systems*, CAS-32(11):1150–1161.

Braess, D. (1986). *Nonlinear Approximation Theory*. Springer Series in Computational Mathematics. Springer-Verlag, Berlin, Germany.

Breiman, L. (1993). Hinging hyperplanes for regression, classification and function approximation. *IEEE Transactions on Information Theory*, 39(3):999–1012.

Bruls, J., Chou, C. T., Haverkamp, B. R. J., and Verhaegen, M. (1997). Linear and non-linear system identification using separable least-squares. Submitted to European Journal of Control.

Bussgang, J. J. (1952). Crosscorrelation functions of amplitude-distorted Gaussian signals. Technical Report 216, MIT Research Laboratory of Electronics.

Cybenko, G. (1989). Approximation by superposition of a sigmoidal function. *Mathematics of control, signals and systems*, 2:303–314.

Dahlquist, G. and Björk, Å. (1974). *Numerical Methods*. Prentice-Hall, Inc, Englewood Cliffs, New Jersey., USA.

de Boor, C. (1978). *A Practical Guide to Splines*, volume 27 of *Applied mathematical sciences*. Springer-Verlag, New York, USA.

de Boor, C. (1992). *Spline Toolbox For Use with MATLAB*. The MathWorks Inc, 24 Prime Park Way, Natick, Mass., USA.

Delyon, B., Juditsky, A., and Benveniste, A. (1995). Accuracy analysis for wavelet approximation. *IEEE Transactions on Neural Networks*, 6(2):332–348.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, pages 1–38.

Dennis, Jr, J. E. and Schnabel, R. B. (1983). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Englewood Cliffs, New Jersey, USA.

Greblicki, W. (1994). Nonparametric identification of Wiener systems by orthogonal series. *IEEE Transactions on Automatic Control*, 39(10):2077–2086.

Hagenblad, A. (1998a). Identifiering av Wienermodeller. In *Reglermöte '98, Preprints*, pages 89–93, Lund, Sweden.

Hagenblad, A. (1999). Initialization and model reduction for Wiener model identification. In *The 7th Mediterranean Conference on Control and Automation*, pages 716–723, Haifa, Israel.

Hagenblad, A. and Ljung, L. (1998). Maximum likelihood identification of Wiener models with a linear regression initialization. In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 712–713, Tampa, Florida, USA.

Hagenblad, J. (1998b). Perinatal mortality in highland cattle. Master's thesis, Lunds universitet, Lund, Sweden.

Hamann, B. and Chen, J.-L. (1994). Data point selection for piecewise linear curve approximation. *Computer Aided Geometric Design*, 11:289–301.

Haverkamp, B. R. J. and Verhaegen, M. (1997). SMI toolbox: State space Model Identification software for multivariable dynamical systems, for use with MATLAB. Technical Report TUD/ET/SCE96.015, Delft University of Technology, The Netherlands.

Haykin, S. (1994). *Neural Networks, A Comprehensive Foundation*. Macmillan, New York, USA.

Hunter, I. W. and Korenberg, M. J. (1986). The identification of nonlinear biological systems: Wiener and Hammerstein cascade models. *Biological Cybernetics*, 55:135–144.

Kalafatis, A., Arifin, N., Wang, L., and Cluett, W. R. (1995). A new approach to the identification of pH processes based on the Wiener model. *Chemical Engineering Science*, 50(23):3693–3701.

Kalafatis, A. D., Wang, L., and Cluett, W. R. (1997). Identification of Wiener-type nonlinear systems in a noisy environment. *International Journal of Control*, 66(6):923–941.

Lindskog, P. (1996). *Methods, Algorithms and Tools for System Identification Based on Prior Knowledge*. PhD thesis, Linköpings universitet, Linköping, Sweden.

Ljung, L. (1978). Convergence analysis of parametric identification methods. *IEEE Transactions of Automatic Control*, AC-23:770–783.

Ljung, L. (1999). *System Identification, Theory for the User*. Prentice Hall, Englewood Cliffs, New Jersey, USA, second edition.

Luenberger, D. G. (1984). *Linear and Nonlinear Programming*. Addison-Wesley, second edition.

Mallat, S. (1998). *A Wavelet Tour of Signal Processing*. Academic Press, San Diego, USA.

MATLAB (1996). *Language Reference Manual, Version 5*. The MathWorks, Inc.

Pajunen, G. A. (1992). Adaptive control of Wiener type nonlinear systems. *Automatica*, 28(4):781–785.

Pucar, P. and Sjöberg, J. (1996). On the parametrization of hinging hyperplanes models. Technical Report Number: LiTH-ISY-R-1831, Linköping University.

Rudin, W. (1976). *Principles of Mathematical Analysis*. McGraw-Hill, third edition.

Schumaker, L. L. and Stanley, S. S. (1996). Shape-preserving knot removal. *Computer Aided Geometric Design*, 13(9):851–872.

Sjöberg, J. (1997). On estimation of nonlinear black-box models: How to obtain a good initialization. In *Proceeding of IEEE Workshop in Neural Networks for Signal Processing, Amelia Island Plantation, Florida, Sep. 24-26*.

Sjöberg, J., Zhang, Q., Ljung, L., Benveniste, A., Delyon, B., Glorennec, P.-Y., Hjalmarsson, H., and Juditsky, A. (1995). Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31(12):1691–1724.

Van Huffel, S. and Vanderwalle, J. (1991). *The Total Least Squares Problem - Computational Aspects and Analysis*. Frontiers in Applied Mathematics. SIAM, Philadelphia, Pennsylvania, USA.

van Overschee, P. and De Moor, B. (1996). *Subspace identification for linear systems – Theory, Implementation, Applications*. Kluwer Academic Publishers.

Wahlberg, B. (1989). Model reductions of high-order estimated models: the asymptotic ML approach. *International Journal of Control*, 49(1):169–192.

Wahlberg, B. (1991). System identification using Laguerre models. *IEEE Transactions on Automatic Control*, 36(5):551–562.

Wahlberg, B. (1994). System identification using Kautz models. *IEEE Transactions on Automatic Control*, 39(6):1276–1282.

Westwick, D. and Verhaegen, M. (1996). Identifying MIMO Wiener systems using subspace model identification methods. *Signal Processing*, 52:235–258.

Wigren, T. (1993). Recursive prediction error identification using the nonlinear Wiener model. *Automatica*, 29(4):1011–1025.

Wigren, T. (1994). Convergence analysis of recursive identification algorithms based on the nonlinear Wiener model. *IEEE Transactions on Automatic Control*, 39(11):2191–2206.

Zhou, K., Doyle, J. C., and Glover, K. (1995). *Robust and Optimal Control*. Prentice Hall, Upper Saddle River, New Jersey, USA.

Zhu, Y. (1998). Identification of Hammerstein models for control. In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 219–220, Tampa, Florida, USA.

Zhu, Y. (1999a). Distillation column identificaiton for control using Wiener model. In *1999 American Control Conference*, Hyatt Regency San Diego, California, USA.

Zhu, Y. (1999b). Parametric Wiener model identification for control. In *14th World Congress of IFAC*, pages 37–42, Beijing, China.

Index

W