Maximum Likelihood Identification of Wiener Models

Anna Hagenblad [a] Lennart Ljung [a] Adrian Wills [b]

[a] annah, ljung@isy.liu.se
Division of Automatic Control,
Linköpings universitet,
SE-581 80 Linköping, Sweden

[b] adrian.wills@newcastle.edu.au
School of Electrical Engineering and Computer Science, University of Newcastle
Callaghan, NSW, 2308, Australia

Abstract

The Wiener model is a block oriented model having a linear dynamic system followed by a static nonlinearity. The dominating approach to estimate the components of this model has been to minimize the error between the simulated and the measured outputs. We show that this will in general lead to biased estimates if there is other disturbances present than measurement noise. The implications of Bussgang's theorem in this context are also discussed. For the case with general disturbances we derive the Maximum Likelihood method and show how it can be efficiently implemented. Comparisons between this new algorithm and the traditional approach confirm that the new method is unbiased and also has superior accuracy.

1 Introduction

So called *block-oriented models* have turned out to be very useful for the estimation of non-linear systems. Such models are built up from linear dynamic systems and nonlinear static mappings in various forms of interconnection. These models are of interest both as reflecting physical realities and as approximations of more general systems. See, e.g. Schoukens et al. (2003) or Hsu et al. (2006) for some general aspects on block-oriented models.

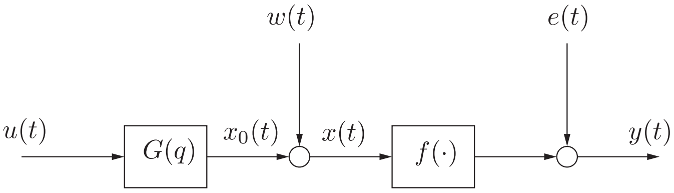


Fig. 1. The Wiener model. The input $u(t)$ and the output $y(t)$ are measurable, but not the intermediate signal $x(t)$. $w(t)$ and $e(t)$ are noise sources. $x_0(t)$ denotes the output of the linear dynamic system G. f is nonlinear and static (memoryless).

The Wiener model, Figure 1, is one such block oriented model, see, e.g., Billings (1980). It describes a system where the first part is linear and dynamic, and the second part, in series with the first, is static and nonlinear. This is a reasonable model for, e.g., a distillation column (Zhu 1999*a*) a pH control process (Kalafatis et al. 1995), biological examples (Hunter & Korenberg 1986), or a linear system with a nonlinear measurement device. If the blocks are multi-variable, it can be shown (Boyd & Chua 1985) that almost any nonlinear system can be approximated arbitrarily well by a Wiener model. In this paper, however, we focus on single input - single output systems.

We will use the notation defined in Figure 1. The input signal is denoted by $u(t)$, the output signal by $y(t)$ and $x(t)$ denotes the intermediate, unmeasurable signal. We will call $w(t)$ process noise and $e(t)$ measurement noise, and assume that they are independent. Note that since G is a linear system, the process noise can equally well be applied anywhere before the nonlinearity with an additional filter.

The Wiener system can be described by the following equations:

$$\begin{aligned} x_0(t) &= G(q,\theta)u(t) \\ x(t) &= x_0(t) + w(t) \\ y(t) &= f\big(x(t),\eta\big) + e(t) \end{aligned} \tag{1}$$

[1] This work was supported by the Swedish Research Council and the Australian Research Council.

This paper will focus on parametric models. We will assume f and G each belongs to a parameterized model class. Examples of such a model class may be polynomials, splines, or neural networks for the nonlinear function f – in general a basis function expansion. The nonlinearity f may also be a piecewise linear function, like a saturation or a dead-zone. Common model classes for G are FIR filters, rational transfer functions (OE models) or state space models, but also for example Laguerre filters may be used.

If the process noise w and the intermediate signal x are unknown, the parameterization of the Wiener model is not unique. A linear block G and a nonlinear block f gives the same complete system as a linear block KG in series with a nonlinear block $f(\frac{1}{K}\cdot)$. (We may also need to scale the process noise variance with a factor K.)

Given input and output data, and model classes for G and f, we want to find (estimate) the parameters θ and η that best match the data, measured as inputs u and outputs y from the system.

2 A Standard Method and Possible Bias Problems

Several different methods to identify Wiener models have been suggested in the literature. A common approach is to parameterize the linear and the nonlinear block, and to estimate the parameters from data, by minimizing an error criterion.

If the process noise $w(t)$ in Figure 1 is disregarded, or zero, a natural criterion is to minimize

$$V_N(\theta,\eta) = \frac{1}{N}\sum_{t=1}^{N}\Big(y(t) - f\big(G(q,\theta)u(t),\eta\big)\Big)^2 \qquad (2)$$

This is a standard approach, and has been used in several papers, e.g., Bai (2003), Westwick & Verhaegen (1996), Wigren (1993). It is also the method for Wiener models, used in available software packages like Ninness & Wills (2006) and Ljung (2007). If the process noise is indeed zero, this is the prediction error criterion. If the measurement noise is white and Gaussian, (2) is also the Maximum Likelihood criterion, see Ljung (1999), and the estimate is thus consistent.

While measurement noise e is discussed in several papers, few consider process noise w. Hunter & Korenberg (1986) is one exception where both the input and the output are subject to noise. Consistency of the estimation method is however not discussed in that paper. It may seem reasonable to use an error criterion like (2) even in the case where there is process noise. However, $f\big(G(q,\theta)u(t),\eta\big)$ is not the true predictor in this case. We will name this method the approximative Prediction Error Method, and we will show that the estimate obtained this way is not necessarily consistent.

2.1 Conditions for Consistency

Suppose that the true system can be described within the model class (cf. Figure 1), i.e., there exist parameters (θ_0,η_0) such that (c.f. Equation (1))

$$y(t) = f\big(G(q,\theta_0)u(t) + w(t),\eta_0\big) + e(t) \qquad (3)$$

An estimate from a certain estimation method is said to be *consistent* if the parameters converge to their true values when the number of data, N, tends to infinity.

To investigate the minimum of the approximative PEM criterion (2) we write the true system as

$$y(t) = f\big(G(q,\theta_0)u(t),\eta_0\big) + \tilde{w}(t) + e(t) \qquad (4)$$

where

$$\tilde{w}(t) = f\big(G(q,\theta_0)u(t) + w(t),\eta_0\big) - f\big(G(q,\theta_0)u(t),\eta_0\big) \qquad (5)$$

We may regard $\tilde{w}(t)$ as a (input-dependent) transformation of the process noise to the output. The stochastic properties such as mean and variance of the process noise will typically not be preserved in the transformation from w to $\tilde{w}$.

Now insert the expression for y in Equation (4) into the criterion (2):

$$\begin{aligned} V_N(\theta,\eta) &= \frac{1}{N}\sum_{t=1}^{N}\Big(f_0 - f + \tilde{w}(t) + e(t)\Big)^2 \qquad (6)\\ &= \frac{1}{N}\sum_{t=1}^{N}\Big(f_0 - f\Big)^2 + \frac{1}{N}\sum_{t=1}^{N}\big(\tilde{w}(t) + e(t)\big)^2 \\ &\quad + \frac{2}{N}\sum_{t=1}^{N}\Big(f_0 - f\Big)\big(\tilde{w}(t) + e(t)\big) \end{aligned}$$

where

$$f_0 \triangleq f\big(G(q,\theta_0)u(t),\eta_0\big), \qquad f \triangleq f\big(G(q,\theta)u(t),\eta\big). \qquad (7)$$

Now, assume all noises are ergodic, so that time averages tend to their mathematical expectations as N tends to infinity. Assume also that u is a (quasi)-stationary sequence, so that is also has well defined sample averages. Let, E denote both mathematical expectation and averaging over time signals (cf. $\bar{E}$ in Ljung (1999)). Using the fact that the measurement noise e is zero mean, and independent of the input u and the process noise w

means that several cross terms will disappear. The criterion then tends to

$$\begin{aligned}\bar{V}(\theta,\eta) = E\Big(f_0 - f\Big)^2 + E\tilde{w}^2(t) + Ee^2(t)\\ + 2E\Big(f_0 - f\Big)\tilde{w}(t)\end{aligned} \qquad (8)$$

The transformed process noise $\tilde{w}$, however, need not be independent of u, so the last term will not disappear.

Note that the criterion (8) has a quadratic form, and the true values (θ_0, η_0) will minimize the criterion if (and essentially only if)

$$E\Big(f\big(G(q,\theta_0)u(t),\eta_0\big) - f\big(G(q,\theta)u(t),\eta\big)\Big)\tilde{w}(t) = 0 \qquad (9)$$

This condition typically does not need to hold, due to the possible dependence between u and $\tilde{w}$. The parameter estimates will thus be biased in general. A concrete example is given in the next section.

2.2 Analytical Solution of a Simple Example

We consider the following system:

$$\begin{aligned}x_0(t) + a_1 x_0(t-1) &= b_1 u(t-1)\\ x(t) &= x_0(t) + w(t)\end{aligned} \qquad (10)$$

followed by a second degree polynomial,

$$\begin{aligned}f\big(x(t)\big) &= c_0 + c_1 x(t) + c_2 x^2(t)\\ y(t) &= f\big(x(t)\big) + e(t)\end{aligned} \qquad (11)$$

For simplicity, assume that the parameters of the linear system, a_1 and b_1, are known and do not need to be estimated. We want to estimate the parameters of the nonlinear subsystem, and will denote the estimates by $\hat{c}_0, \hat{c}_1$ and $\hat{c}_2$.

For this simple example, the analytical minimum of the criterion (2) can be calculated. Since the parameters of the linear subsystem are known, so is the output x_0. We can then write the predicted output (using the approximate predictor in (2)) as

$$\begin{aligned}\hat{y}(t) = f(G(q,\theta)u(t),\eta) &= f(x_0(t),\eta)\\ &= \hat{c}_0 + \hat{c}_1 x_0(t) + \hat{c}_2 x_0^2(t)\end{aligned} \qquad (12)$$

We will assume all signals, noises as well as inputs, are Gaussian, zero mean and ergodic. We use λ_x to denote the variance of x_0, while the variance of w is denoted by λ_w, and the variance of e by λ_e. As N tends to infinity, the criterion (2) tends to the limit (8)

$$\begin{aligned}\bar{V} =& E(y-\hat{y})^2 = E\big(c_0 + c_1(x_0 + w) + c_2(x_0+w)^2\\ &+ e - \hat{c}_0 - \hat{c}_1 x_0 - \hat{c}_2 x_0^2\big)^2\\ =& E\big((c_2 - \hat{c}_2)x_0^2 + (c_1 - \hat{c}_1)x_0\\ &+ c_0 - \hat{c}_0 + 2c_2 x_0 w + c_2 w^2 + c_1 w + e\big)^2\end{aligned}$$

Since all signals are Gaussian, independent and zero mean, all odd terms will be zero. The fourth order moments are $Ex^4 = 3\lambda_x^2$ and $Ew^4 = 3\lambda_w^2$. What remains is

$$\begin{aligned}\bar{V} =& 3(c_2 - \hat{c}_2)^2\lambda_x^2 + (c_1 - \hat{c}_1)^2\lambda_x + (c_0 - \hat{c}_0)^2\\ &+ 4c_2\lambda_x\lambda_w + 3c_2^2\lambda_w^2 + c_1^2\lambda_w + \lambda_e + 2(c_0 - \hat{c}_0)\\ &\times (c_2 - \hat{c}_2)\lambda_x + 2c_2(c_2 - \hat{c}_2)\lambda_x\lambda_w + 2c_2(c_0 - \hat{c}_0)\lambda_w\end{aligned}$$

The minimum with respect to $\hat{c}_i$ can be found by setting the gradient equal to zero, which gives the equation system

$$\begin{aligned}(c_0 - \hat{c}_0) + (c_2 - \hat{c}_2) + c_2\lambda_w &= 0\\ (c_1 - \hat{c}_1)\lambda_x &= 0\\ 3(c_2 - \hat{c}_2)\lambda_x^2 + (c_0 - \hat{c}_0)\lambda_x + 3c_2\lambda_x\lambda_w &= 0\end{aligned}$$

with the solution

$$\begin{aligned}\hat{c}_0 &= c_0 + c_2\lambda_w\\ \hat{c}_1 &= c_1\\ \hat{c}_2 &= c_2\end{aligned}$$

The estimate of c_0 will thus be biased. This result is also confirmed by the results of the simulations in Section 5, where the parameters $\hat{c}_1$ and $\hat{c}_2$ are estimated consistently, and the estimate of $\hat{c}_0$ is close to the true value c_0 plus the variance of the process noise.

Similar results can also be derived for other nonlinearities.

2.3 Bussgang's Theorem and its Implication for Wiener Models

Bussgang's theorem (Bussgang 1952) says the following:

Theorem 1 ***[Bussgang]*** *Let $m(t)$ and $n(t)$ be two real-valued, jointly Gaussian stationary processes. Let $f(\cdot)$ be a nonlinear function and let the stochastic process $g(t)$ be defined by*

$$g(t) = f(n(t))$$

Then the cross spectrum between m and n, $\Phi_{mn}(\omega)$, is proportional to the cross spectrum between m and g:

$$\Phi_{mg}(\omega) = \kappa\Phi_{mn}(\omega) \qquad (13)$$

where κ is a real-valued constant (that may be zero.)

This theorem has been applied to the estimation of Wiener model by many authors, e.g. (Westwick & Verhaegen 1996), (Greblicki 1994) and Hunter & Korenberg (1986). It can be used to obtain a good estimate of the linear part of the model. It is interesting to note that the result applies also to our more general situation with process noise w as in Figure 1. In fact, we have the following Lemma:

Lemma 1 *Consider the model structure defined by Figure 1. Assume that the the input $u(t)$ and the process noise $w(t)$ are independent, Gaussian, stationary processes (not necessarily white). Assume that the measurement noise $e(t)$ is a stationary stochastic process, independent of u and w. It is however not assumed that e is neither white nor Gaussian. Let $G(q, \theta)$ be an arbitrary transfer function parameterization with freely adjustable gain, such that $G(q, \theta_0) = G_0(q)$ (the true linear part of the system) for some parameter value θ_0. Let θ be estimated from u and y using an output error method, neglecting any possible presence of a nonlinearity:*

$$\hat{\theta}_N = \underset{\theta}{\operatorname{argmin}} \sum_{t=1}^{N} (y(t) - G(q, \theta)u(t))^2 \tag{14}$$

Then

$$G(q, \hat{\theta}_N) \to \kappa G_0(q) \quad \text{as} \quad N \to \infty \tag{15}$$

for some real constant κ (that may be zero).

Proof: Define

$$x_0(t) = G_0(q)u(t); \qquad x(t) = x_0(t) + w(t) \tag{16}$$

$$y_0(t) = f(x(t)); \qquad y(t) = y_0(t) + e(t) \tag{17}$$

Then since e is independent of u the cross spectra between u and y, y_0 will be the same: $\Phi_{yu} = \Phi_{y_0 u}$. Since u, and w are Gaussian, then $x(t)$ is Gaussian, so Bussgang's theorem tells us that $\Phi_{y_0 u} = \kappa \Phi_{xu}$ since $y_0 = f(x)$. Moreover, since w is independent of u, $\Phi_{xu} = \Phi_{x_0 u}$. The resulting conclusion is that

$$\Phi_{yu}(\omega) = \kappa \Phi_{x_0 u}(\omega) = \kappa G_0(e^{i\omega})\Phi_u(\omega) \tag{18}$$

Now, it is well known (see e.g. Chapter 8 in Ljung 1999) that $\hat{\theta}_N$ will converge to a value that minimizes

$$\begin{aligned} V(\theta) =& E(y(t) - G(q, \theta)u(t))^2 = \\ & \frac{1}{2\pi} \int \big(\Phi_y(\omega) - 2Re[G(e^{-i\omega}, \theta)\Phi_{yu}(\omega)] \\ & + |G(e^{i\omega}, \theta)|^2 \Phi_u(\omega)\big) d\omega \end{aligned}$$

Insert (18) into this expression, and replace the θ-independent term Φ_y with the θ-independent term $\kappa^2 |G_0(e^{i\omega})|^2 \Phi_u(\omega)$. Minimizing $V(\theta)$ is thus the same as minimizing

$$\begin{aligned} W(\theta) &= \int \Big(\kappa^2 |G_0(e^{i\omega})|^2 \Phi_u(\omega) - 2Re[\kappa G(e^{-i\omega}, \theta) \\ &\quad \times G_0(e^{i\omega})\Phi_u(\omega)] + +|G(e^{i\omega}, \theta)|^2 \Phi_u(\omega)\Big) d\omega \\ &= \int |\kappa G_0(e^{i\omega}) - G(e^{i\omega}, \theta)|^2 \Phi_u(\omega) d\omega \end{aligned} \tag{19}$$

which proves the lemma. ■

The theorem is a consequence of the fact that the best linear system approximation (cf Ljung 2001) that relates u to y is proportional to the linear part G_0 of the true system.

Basically this means that an estimate of the linear system $G(q)$ will be consistent (up to the gain) for many other common linear identification methods. Note that the gain of G cannot be estimated anyway, since a gain factor can be moved between G and f without affecting the input-output behavior.

Remark 1: It is essential that no noise model is built simultaneously with estimating G. The best linear description of the noise will be pretty complicated, since all the nonlinear effects are pushed to the residuals. This means that ARMAX, ARX and state-space models in innovations form that have common dynamics between noise and input, will not give an input dynamics model subject to (15).

Remark 2: In line with the previous remark it should be noted that independence between the noise model and the input dynamics model is essential also when using subspace methods, like N4SID (e.g. Van Overschee & DeMoor 1996). If such methods employ prediction horizons with past outputs, biased estimates of the linear dynamics will results. (More precisely, in the language of the system identification toolbox, (Ljung 2007), the property `N4HORIZON` must be of the form `[r,0,su]`.) This is in accordance with the use of the Output-Error method MOESP as a subspace method in (Westwick & Verhaegen 1996).

Having found G means that we know x_0 (up to scaling). If there is no process noise w so $x(t) = x_0(t)$, it is a simple problem to estimate the static nonlinearity $y(t) = f(x(t)) + e(t)$ from y and x.

However, if w is non-zero, the remaining problem to estimate f is still non-trivial: Find f from x_0 and y, where

$$y(t) = f(x_0(t) + w(t)) + e(t) \tag{20}$$

This is a nonlinear regression problem with disturbances affecting the regressors. The estimate of the parameters of f need not be consistent if simple methods are applied, as seen in Section 2.2. This is further illustrated in the simulations in Section 5.

3 Maximum Likelihood Estimation

3.1 *Derivation of the Likelihood Function for White Disturbances*

The likelihood function is the probability density function (PDF) of the outputs $y^N = \{y(1), y(2), ..., y(N)\}$ for given parameters θ and η. We shall also assume that the input sequence $u^N = \{u(1), u(2), ...u(N)\}$ is a given, deterministic sequence. (Alternatively, we condition the PDF wrt to this sequence, if it is described as a stochastic process.) Let $p_{y^N}(\theta, \eta; u^N)$ denote this PDF. For an observed data set y_*^N, the ML estimate is the one maximizing the likelihood function:

$$(\hat{\theta}, \hat{\eta}) = \underset{\theta,\eta}{\operatorname{argmax}}\, p_{y^N}(\theta, \eta; Z_*^N) \tag{21}$$

where $Z_*^N = \{u^N, y_*^N\}$.

For the Wiener model (Figure 1) we first assume that the disturbance sequences $e(t)$ and $w(t)$ are white noises. This means that for given u^N, $y(t)$ will also be a sequence of independent variables. This in turn implies that the PDF of y^N will be the product of the PDFs of $y(t), t = 1, \dots, N$. It is thus sufficient to derive the PDF of $y(t)$. To simplify notation we shall use $y(t) = y, x(t) = x$ for short.

To find the PDF, we introduce the intermediate signal x as a nuisance parameter. The PDF of y given x is basically a reflection of the PDF of e, since $y(t) = f\big(x(t)\big) + e(t)$ It is easy to find if e is white noise:

$$p_y(y|x) = p_e\big(y - f(x, \eta)\big) \tag{22}$$

where p_e is the PDF of e.

The same is true for the PDF of x given u^N if w is white noise:

$$x(t) = G(q, \theta)u(t) + w(t) = x_0(t, \theta) + w(t) \tag{23}$$

With given u^N and θ, x_0 is a known, deterministic variable, so

$$p_x(x|u^N, \theta) = p_w\big(x - x_0(\theta)\big) = p_w\big(x - G(q, \theta)u(t)\big) \tag{24}$$

where p_w is the PDF of w.

Now by integrating over all $x \in \mathbf{R}$, we then eliminate this unmeasurable signal from our equations:

$$\begin{aligned} p_y(\theta, \eta; Z_*^N) &= \int_{x\in\mathbf{R}} p_{x,y}(x, y|\theta, \eta; u^N) dx = \\ &= \int_{x\in\mathbf{R}} p_{y|x}(y|\theta, \eta, x; u^N)\, p_x(x|\theta, \eta; u^N) dx = \\ &= \int_{x\in\mathbf{R}} p_e\big(y - f(x, \eta)\big)\, p_w\big(x - G(q, \theta)u(t)\big) dx \end{aligned} \tag{25}$$

We now assume that the process noise $w(t)$ and the measurement noise $e(t)$ are Gaussian, with zero means and variances λ_w and λ_e respectively, i.e.

$$\begin{aligned} p_e\big(\epsilon(t)\big) &= \frac{1}{\sqrt{2\pi\lambda_e}} e^{-\frac{1}{2\lambda_e}\epsilon^2(t)} \quad \text{and} \\ p_w\big(v(t)\big) &= \frac{1}{\sqrt{2\pi\lambda_w}} e^{-\frac{1}{2\lambda_w}v^2(t)} \end{aligned} \tag{26}$$

for each time instant t. Since the noise is white, the joint likelihood is the product over all time instants, and thus

$$\begin{aligned} p_y(y^N|\theta, \eta; u^N) &= \left(\frac{1}{2\pi\sqrt{\lambda_e\lambda_w}}\right)^N \prod_{t=1}^{N} \int_{-\infty}^{\infty} e^{-\frac{1}{2}E(t,\theta,\eta)}\, dx(t) \\ &= \left(\frac{1}{2\pi\sqrt{\lambda_e\lambda_w}}\right)^N \int_{x(1)=-\infty}^{\infty} \cdot \int_{x(2)=-\infty}^{\infty} \cdots \\ &\cdot \int_{x(N)=-\infty}^{\infty} e^{-\frac{1}{2}\sum_{t=1}^{N} E(t,\theta,\eta)}\, dx^N \end{aligned} \tag{27}$$

where

$$E(t, \theta, \eta) = \frac{1}{\lambda_e}\Big(y(t) - f\big(x(t), \eta\big)\Big)^2 + \frac{1}{\lambda_w}\big(x(t) - G(q, \theta)u(t)\big)^2 \tag{28}$$

Given data $Z_*^N = \{u_*^N, y_*^N\}$, we can calculate p_y and its gradients for each θ and η. This means that the ML criterion (21) can be maximized numerically.

We may also note that each integral in (27) depends on $x(t)$ for only one time instant t, so they can be computed in parallel.

If the noise covariances λ_w and λ_e are unknown, they can just be included among the parameters θ and η and their ML estimates are still obtained by (21).

The derivation of the Likelihood function appeared in Hagenblad & Ljung (2000).

3.2 *Special Cases: No Process Noise or No Measurement Noise*

Most approaches suggested in the literature restrict the noise to either process noise or measurement noise. In these cases, the likelihood function (27) is considerably simplified, and the criterion is reduced to something we recognise from other references.

First, the case of no process noise, $\lambda_w = 0$. Since the only stochastic part is the measurement noise, we have

$$\begin{aligned} p_y(\theta,\eta;Z_*^N) &= p_e\Big(y(t) - f\big(G(q,\theta)u(t),\eta\big)\Big) \\ &= \prod_{t=1}^{N} \frac{1}{\sqrt{2\pi\lambda_e}} e^{-\frac{1}{2\lambda_e}\Big(y(t)-f\big(G(q,\theta)u(t),\eta\big)\Big)^2} \end{aligned} \tag{29}$$

Maximizing this is equivalent to minimizing the criterion

$$V_N(\theta,\eta) = \frac{1}{N}\sum_{t=1}^{N}\Big(y(t) - f\big(G(q,\theta)u(t),\eta\big)\Big)^2 \tag{30}$$

This is the prediction error criterion (2) discussed before. It gives a consistent estimate if the condition of no process noise is satisfied. However, if there is process noise, this criterion does not use the true predictor, and may give biased estimates, as was discussed in Section 2.

For systems with no measurement noise, $\lambda_e = 0$ the criterion (27) forces $y(t) = f(x(t),\eta)$. So if f is invertible, the ML criterion becomes

$$\begin{aligned} p_y(\theta,\eta;Z_*^N) &= p_w\Big(f^{-1}\big(y(t),\eta\big) - G(q,\theta)u(t)\Big) \\ &= \prod_{t=1}^{N} \frac{1}{\sqrt{2\pi\lambda_w}} e^{-\frac{1}{2\lambda_w}\Big(f^{-1}\big(y(t),\eta\big)-G(q,\theta)u(t)\Big)^2} \end{aligned} \tag{31}$$

The maximum can be found by maximizing the logarithm, which reduces the problem to minimizing the criterion

$$V_N(\theta,\eta) = \frac{1}{N}\sum_{t=1}^{N}\Big(f^{-1}\big(y(t),\eta\big) - G(q,\theta)u(t)\Big)^2 \tag{32}$$

which is the criterion discussed by, e.g., (Kalafatis et al. 1995), (Zhu 1999*b*).

3.3 Colored Noises

If the process and/or measurement noise is colored, we may represent the Wiener system as in Figure 2.

The following equations give the output:

$$\begin{aligned} x(t) &= G(q,\theta)u(t) + H_w(q,\theta)w(t) \\ y(t) &= f\big(x(t),\eta\big) + H_e(q,\eta)e(t) \end{aligned} \tag{33}$$

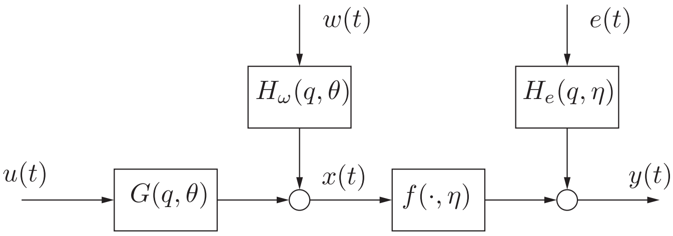


Fig. 2. Wiener model with colored noises. Both $w(t)$ and $e(t)$ are still supposed to be white noise sources, which are filtered through $H_w(q,\theta)$ and $H_e(q,\eta)$, respectively.

By using predictor form, see (Ljung 1999), we may write this as

$$\begin{aligned} x(t,u^N,\theta) &= H_w^{-1}(q,\theta)G(q,\theta)u(t) + \big(1 - H_w^{-1}(q,\theta)\big)x(t) \\ &\quad + w(t) \triangleq \hat{x}(t|x^{t-1},u^N,\theta) + w(t) \\ y(t) &= H_e^{-1}(q,\eta)f\big(x(t),\eta\big) + \big(1 - H_e^{-1}(q,\eta)\big)y(t) \\ &\quad + e(t) \triangleq \hat{y}(t|y^{t-1},x^N(u^N,\theta),\eta) + e(t) \end{aligned} \tag{34}$$

The only stochastic parts are e and w. For a given sequence x^N, the joint PDF of y^N is obtained in the standard way, cf eq (5.74), Lemma 5.1, in Ljung (1999):

$$p_{y^N}(y^N|x^N) = \prod_{t=1}^{N} p_e(y(t) - \hat{y}(t|y^{t-1},x^N(u^N,\theta),\eta)) \tag{35}$$

By the same calculations, the joint PDF for x^N is

$$p_{x^N}(x^N) = \prod_{t=1}^{N} p_w(x(t) - \hat{x}(t|y^{t-1},u^N,\theta)) \tag{36}$$

The likelihood function for y^N is thus obtained from (35) by integrating out the nuisance parameter x^N using its PDF (36):

$$\begin{aligned} &p_{y^N}(y^N,\theta,\eta;u^N) = \\ &\qquad = \int_{x(1)\in\mathbf{R}}\int_{x(2)\in\mathbf{R}}\cdots\int_{x(N)\in\mathbf{R}} \\ &\prod_{t=1}^{N} p_e\big(H_w^{-1}(q,\theta)x(t) - H_w^{-1}(q,\theta)G(q,\theta)u(t)\big) \times \\ &p_w\Big(H_e^{-1}(q,\eta)y(t) - H_e^{-1}(q,\eta)f\big(x(t),\eta\big)\Big)\, dx^N \end{aligned} \tag{37}$$

In the case e and w are Gaussian, we obtain

$$\begin{aligned} &\prod_{t=1}^{N} p_e\big(H_w^{-1}(q,\theta)\left[x(t) - G(q,\theta)\right]u(t)\big) \\ &\qquad \times p_w\Big(H_e^{-1}(q,\eta)\big[y(t) - f\big(x(t),\eta\big)\big]\Big) \\ &\qquad\qquad = e^{-\frac{1}{2}\sum_{t=1}^{N} E(t,\theta,\eta)} \end{aligned} \tag{38}$$

similar to (27), where, this time,

$$E(t,\theta,\eta) = \\ \frac{1}{\lambda_w}\big(H_w^{-1}(q,\theta)\,[x(t)-G(q,\theta)]\,u(t)\big)^2 \\ +\frac{1}{\lambda_e}\Big(H_e^{-1}(q,\eta)\,\big[y(t)-f\big(x(t),\eta\big)\big]\Big)^2 \tag{39}$$

Notice that this time filtered versions of $x(t)$ enter the integral, so the integration is a true multi-integral over all sequences x^N. This is hardly doable by direct integration in practise, unless the inverse noise filters are short FIR filters. It would then be interesting to evaluate the integration over x^N by probabilistic techniques.

Remark: There is a conceptual relation with the EM-algorithm in this formulation. In a sense, the signal x is a "missing signal", which makes the formulation of the likelihood function easy. The EM-algorithm would alternate between estimating this signal x and maximizing the likelihood. In our algorithm, instead, x is seen as a nuisance parameter that is integrated out in the criterion. This seems to be a more effective approach for the current problem.

4 Implementation

It was mentioned in Section 3 that numerical methods could be used to evaluate the likelihood integral in Equation (27), which could in turn be used as part of an iterative search procedure to find for the maximum likelihood estimate. While this may appear intractable, this section describes a practical algorithm for achieving the above.

In particular, we use a gradient-based iterative search method combined with numerical integration to form the ML estimate. The algorithm is profiled against the approximative PEM in Section 2 where the results from several Monte-Carlo simulations are discussed. It should be noted that the computation time for this algorithm is relatively modest and can easily be carried out on standard desktop computers.

In order to avoid numerical conditioning problems, we consider the equivalent problem of minimizing the negative log-likelihood function provided below.

$$(\hat\theta,\hat\eta,\hat\lambda_w,\hat\lambda_e) = \underset{\theta,\eta,\lambda_w,\lambda_e}{\operatorname{argmin}}\ L(\theta,\eta,\lambda_w,\lambda_e) \tag{40}$$

where

$$\begin{aligned} L(\theta,\eta,\lambda_w,\lambda_e) &\triangleq -\log\big(p_y(\theta,\eta,\lambda_w,\lambda_e;Z_*^N)\big)\\ &= N\log(2\pi)+\frac{N}{2}\log(\lambda_w\lambda_e)\\ &\quad -\sum_{t=1}^{N}\log\left(\int_{-\infty}^{\infty}e^{-\frac{1}{2}E(t,\theta,\eta)}dx\right) \end{aligned} \tag{41}$$

and $E(t)$ is given by Equation (28).

First, it must be noted that the criterion certainly is non-convex and may have several local minima. There is always a risk to get trapped in a nonglobal, local minimim, and therefore the initial parameter value for the search must be chosen with some care. This is a problem that (40) shares with likelihood functions in general for dynamic systems.

In order to solve (40) we use an iterative gradient-based approach. This approach constructs a local model of the function L using derivative information; the method then minimizes the local model in the hope that the model is a close match to L and will thus minimize L. Since the model is rarely good enough to minimize L in a single step, search strategies use the local solution as "search direction", and then the function L is reduced along the search direction to obtain a new parameter estimate. More precisely, let

$$\vartheta \triangleq \Big[\theta^T\ \eta^T\ \lambda_w\ \lambda_e\Big]^T \tag{42}$$

then at iteration k, $L(\vartheta_k)$ is modeled locally as

$$L(\vartheta_k+p)\approx L(\vartheta_k)+g_k^Tp+\frac{1}{2}p^TH_k^{-1}p, \tag{43}$$

where g_k is the derivative of L with respect to ϑ evaluated at ϑ_k, i.e.

$$g_k \triangleq \left.\frac{\partial L(\vartheta)}{\partial\vartheta}\right|_{\vartheta=\vartheta_k}, \tag{44}$$

and H_k^{-1} is some symmetric matrix. If a Newton direction is desired, then H_k^{-1} would be the inverse of Hessian matrix, but the Hessian matrix itself may be quite expensive to compute. Due to this, here we employ a quasi-Newton method where H_k is updated at each iteration based on local gradient information so that it resembles the Hessian matrix in the limit. In particular, we use the well-known BFGS update strategy (Nocedal & Wright 2006, Section 6.1), which guarantees that H_k is positive definite and symmetric so that

$$p_k = -H_kg_k \tag{45}$$

minimizes (43). The new parameter estimate ϑ_{k+1} is then obtained by updating the previous one via

$$\vartheta_{k+1} = \vartheta_k + \alpha_k p_k, \tag{46}$$

where α_k is selected such that

$$L(\vartheta_k + \alpha_k p_k) < L(\vartheta_k). \tag{47}$$

With this as background, the algorithm used in this paper proceeds as shown in Algorithm 1, which is adopted from (Nocedal & Wright 2006, Algorithm 8.1).

Note that in step 6 of the algorithm, where new parameter values are calculated, it is important to ensure that the system is well-defined; i.e., that G is stable, that f is well-defined, and that the variance estimates are positive.

Algorithm 1: Gradient-based search for ML estimate

Given ϑ_0, set $k = 0$ and iterate the following steps.

```
(1) Compute the gradient vector $g_k$.
(2) IF $k = 0$, set $H_0 = \frac{0.01}{||g_0||_2} I$. ENDIF.
(3) IF $k > 0$,
    (a) Compute
        $\delta_k = \vartheta_k - \vartheta_{k-1},\ \gamma_k = g_k - g_{k-1},\ \rho_k = \frac{1}{\delta_k^T \gamma_k}.$   (48)
    (b) IF $k = 1$, set $H_0 = \frac{\delta_k^T \gamma_k}{\gamma_k^T \gamma_k} I$. ENDIF.
    (c) Compute $H_k$ via
        $H_k = \left(I - \rho_k \delta_k \gamma_k^T\right) H_{k-1} \left(I - \rho_k \gamma_k \delta_k^T\right) + \rho \delta_k \delta_k^T.$   (49)
    ENDIF.
(4) Compute $p_k = -H_k g_k$.
(5) Set $\alpha_k = 1$.
(6) WHILE $L(\vartheta_k + \alpha_k p_k) \geq L(\vartheta_k) + 10^{-6} \alpha_k g_k^T p_k$,
        set $\alpha_k \leftarrow 0.5 \alpha_k$.
      ENDWHILE.
(7) Set $\vartheta_{k+1} = \vartheta_k + \alpha_k p_k$.
(8) IF converged, then STOP, ENDIF.
(9) Set $k = k + 1$ and goto step 1.                ■
```

It is essential for the above algorithm that we can evaluate the cost function $L(\vartheta_k)$ and its derivatives g_k, where the i'th element of g_k, denoted $g_k(i)$, is given by

$$g_k(i) = \left[\frac{N}{2} \frac{\partial \log(\lambda_w)}{\partial \vartheta(i)} + \frac{N}{2} \frac{\partial \log(\lambda_w)}{\partial \vartheta(i)} + \frac{1}{2} \sum_{t=1}^{N} \frac{\int_{-\infty}^{\infty} \frac{\partial E(t,\theta,\eta)}{\partial \vartheta(i)} e^{-\frac{1}{2} E(t,\theta,\eta)} dx}{\int_{-\infty}^{\infty} e^{-\frac{1}{2} E(t,\theta,\eta)} dx} \right] \Bigg|_{\vartheta = \vartheta_k} \tag{50}$$

This in turn requires that we compute the integrals in (41) and (50). Note, that the exponential term $\exp(-\frac{1}{2} E(t,\theta,\eta))$ appears in both these integrals, and the derivatives of $E(t,\theta,\eta)$ with respect to θ and η can be computed prior to evaluating the integral.

In general, evaluating these integrals will amount to approximating them via numerical integration methods, which is the approach used in this paper. In particular, we employ a fixed-interval grid over x and use the composite Simpson's rule to obtain the approximation (Press et al. 1992, Chapter 4). More generally however, the reason for using a fixed grid (not necessarily of fixed-interval as used here) is that it allows straightforward computation of $L(\vartheta_k)$ and its derivative g_k at the same grid point. Hence, a more elaborate approach might employ an adaptive numerical integration method that ensures the same grid points in calculating $L(\vartheta_k)$ and g_k.

Algorithm 2 details this computation and generates a number $\bar{L}$ and a vector $\bar{g}$ such that $L(\vartheta) \approx \bar{L}$ and $g(\vartheta) \approx \bar{g}$. For clarity, the algorithm is written as iterations over t and j, but these steps are not interdependent, and can be computed in parallel. The algorithm can also be extended to compute the Hessian if desired.

Algorithm 2:Numerical computation of likelihood and derivatives

Given an odd number of grid points M, the parameter vector ϑ and the data Z_*^N, perform the following steps.

NOTE: After the algorithm terminates, $L(\vartheta) \approx \bar{L}$ and $g(\vartheta) \approx \bar{g}$.

```
(1) Simulate the system $x_0(t) = G(\theta, q) u(t)$.
(2) Specify grid vector $\Delta \in \mathbb{R}^M$ as $M$ equidistant
    points between the limits $[a\ b]$, so that $\Delta(1) = a$ and
    $\Delta(i+1) = \Delta(i) + (b-a)/M$ for all $i = 1, \ldots, M-1$.
(3) Set $\bar{L} = N \log(2\pi) + \frac{N}{2} \log(\lambda_w \lambda_e)$, and $\bar{g}(i) = 0$ for
    $i = 1, \ldots, n_\vartheta$.
(4) FOR t=1:N,
    (a) FOR j=1:M, compute
            $x = x_0(t) + \Delta(j),$   (51)
            $\alpha = x - x_0(t),$   (52)
            $\beta = y(t) - f(x, \eta),$   (53)
            $\gamma_j = e^{-\frac{1}{2}(\alpha^2/\lambda_w + \beta^2/\lambda_e)},$   (54)
            $\delta_j(i) = \gamma_j \frac{\partial E(t)}{\partial \vartheta(i)}, \qquad i = 1, \ldots, n_\vartheta,$   (55)
        ENDFOR
```

(b) Compute

$$\begin{aligned}\kappa =&\frac{(b-a)}{3M}\left(\gamma_1 + 4\sum_{j=1}^{\frac{M-1}{2}}\gamma_{2j}\right.\\ &\left.+2\sum_{j=1}^{\frac{M-3}{2}}\gamma_{2j+1}+\gamma_M\right),\\ \pi(i) =&\frac{(b-a)}{3M}\left(\delta_1(i) + 4\sum_{j=1}^{\frac{M-1}{2}}\delta_{2j}(i)\right.\\ &\left.+2\sum_{j=1}^{\frac{M-3}{2}}\delta_{2j+1}(i)+\delta_M(i)\right), \quad i=1,\dots,n_\vartheta,\end{aligned}$$

$$\begin{aligned}\bar{L} &= \bar{L} - \log(\kappa),\\ \bar{g}(i) &= \bar{g}(i) + \frac{1}{2}\left(\frac{\partial \log(\lambda_w\lambda_e)}{\partial\vartheta(i)} + \frac{\pi(i)}{\kappa}\right),\\ i &= 1,\dots,n_\vartheta,\end{aligned}$$

ENDFOR

■

As a final note, we point out that the derivatives of $E(t)$ can reuse the calculation of α and β (from Algorithm 2) since

$$\begin{aligned}\frac{\partial E(t)}{\partial\theta(i)} &= -\frac{2}{\lambda_w}\frac{\partial G(q,\theta)u(t)}{\partial\theta(i)}(x - G(q,\theta)u(t))\\ &= -\frac{2\alpha}{\lambda_w}\frac{\partial G(q,\theta)u(t)}{\partial\theta(i)}\\ \frac{\partial E(t)}{\partial\eta(i)} &= -\frac{2}{\lambda_e}\frac{\partial f(x,\eta)}{\partial\eta(i)}(y(t) - f(x,\eta)u(t))\\ &= -\frac{2\beta}{\lambda_e}\frac{\partial f(x,\eta)}{\partial\eta(i)}\\ \frac{\partial E(t)}{\partial\lambda_w} &= -\frac{\alpha^2}{\lambda_w^2}\\ \frac{\partial E(t)}{\partial\lambda_e} &= -\frac{\beta^2}{\lambda_e^2}\end{aligned}$$

5 Simulation study

Two different systems were simulated, and the parameters were estimated using the two different methods described in the paper, namely the approximative PEM described in Section 2, and the ML method described in Section 3. In both cases, 1000 data sets of 1000 data points each were generated in a Monte-Carlo simulation. The prediction error criterion was minimized using the UNIT toolbox, (Ninness & Wills 2006). To help avoid possible local minima, the search was initialized using the true values.

The ML implementation is described in Section 4. In addition to the system parameters, the noise covariances λ_w and λ_e were estimated. The parameter search was initialized with the results from the approximative PEM. The limits for the integration $[a, b]$ (see Algorithm 2) were selected as $\pm 6\sqrt{\lambda_w}$, which corresponds to a confidence interval of 99.9999 % for the signal $x(t)$. The number of grid points was chosen to be 1001.

5.1 *Example 1: A first order dynamic system with polynomial nonlinearity*

The first example has a first order linear system,

$$\begin{aligned}x_0(t) + a_1x_0(t-1) &= b_1u(t-1)\\ x(t) &= x_0(t) + w(t)\end{aligned} \tag{56}$$

followed by a second degree polynomial,

$$\begin{aligned}f\big(x(t)\big) &= c_0 + c_1x(t) + c_2x^2(t)\\ y(t) &= f\big(x(t)\big) + e(t)\end{aligned} \tag{57}$$

This is the same example as was investigated analytically in Section 2.2.

The model structure and the degree of the polynomial are assumed to be known. We use white Gaussian noise with variance 1 as input u. The process noise w is also Gaussian with variance 4, and the measurement noise e is Gaussian with variance 1. These three signals are mutually independent. To get a unique solution, we fix the coefficient for $u(t-1)$, b_1, to 1 in the estimation. The parameters to estimate are thus the coefficient for $x_0(t-1)$, namely a_1, and the coefficients of the polynomial, c_0, c_1 and c_2. The true values are given in Table 1, together with the estimates for the PEM and the ML method. Histograms of the results are shown in Figure 3.

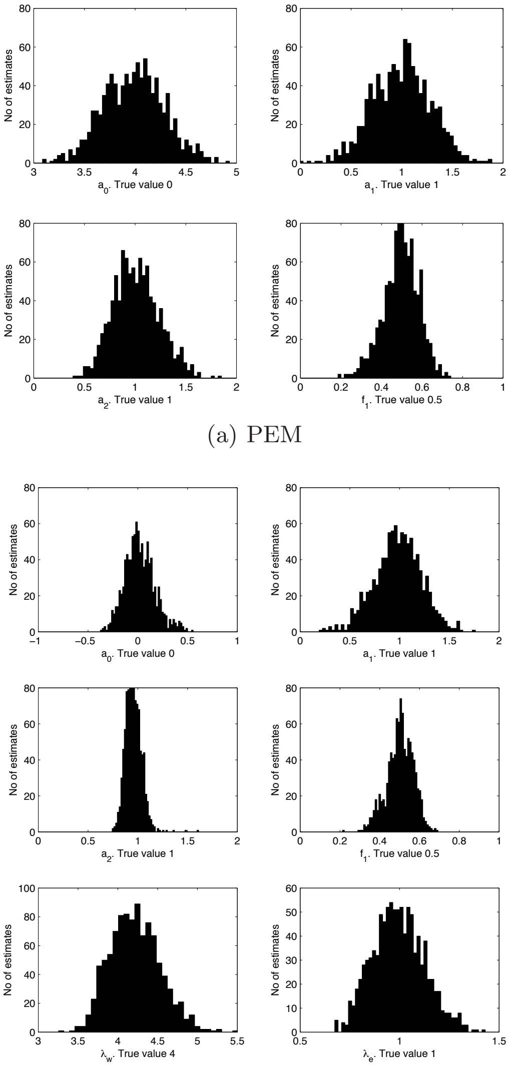
The results confirm the theoretical analyses: The parameters of the linear subsystem are estimated consistently with both the approximative PEM and the ML method. The nonlinear parameter c_0 is biased for the approximative PEM, while the ML method estimates all parameters consistently. The ML method also estimates the noise variances successfully. In this example, the variance of the PEM estimates is larger than the variance of the ML estimate.

The computation time for the ML estimate is reasonable, we used an Intel based laptop with 2.33GHz processor. One estimation run needed 50 iterations on average, with about 0.2 seconds per iteration, thus requiring 10 seconds per estimation.

Example 1

Par	True	Approx. PEM	ML
c_0	0	3.9857 ± 0.3006	0.0220 ± 0.1509
c_1	1.0000	0.9975 ± 0.2870	0.9827 ± 0.2369
c_2	1.0000	1.0136 ± 0.2287	0.9576 ± 0.0830
a_1	0.5000	0.4957 ± 0.0882	0.5032 ± 0.0671
λ_w	4.0000	n.e.	4.2125 ± 0.3378
λ_e	1.0000	n.e.	0.9952 ± 0.1302

Table 1
Parameter estimates with standard deviations for Example 1, using approximative PEM and ML. The mean and standard deviations are computed over 1000 runs. The notation n.e. stands for "not estimated" as the noise variances are not estimated with the approximative PEM.



(a) PEM

(b) ML

Fig. 3. Example 1. Histograms over parameter estimates from 1000 Monte Carlo simulations. a): Using the approximative PEM. b): Using the ML method

5.2 *Example 2: Second order dynamical system with saturation*

Our second example is a second order system with complex poles, followed by a saturation. The input u and process noise w are Gaussian, with zero mean and variance 1, while the measurement noise e is Gaussian with zero mean and variance 0.1. The system is given by

$$\begin{aligned} x_0(t) + a_1 x_0(t-1) + a_2 x_0(t-2) \\ = u(t) + b_1 u(t-1) + b_2 u(t-2) \\ x(t) = x_0(t) + w(t) \\ f\big(x(t)\big) = \begin{cases} c_1 & \text{for } x(t) \le c_1 \\ x(t) & \text{for } c_1 < x(t) \le c_2 \\ c_2 & \text{for } c_2 < x(t) \end{cases} \\ y(t) = f\big(x(t)\big) + e(t) \end{aligned} \tag{58}$$

Here, we estimate the parameters $a_1, a_2, b_1, b_2, c_1, c_2$. Again, a Monte-Carlo simulation with 1000 data sets were generated. The true values of the parameters, and the results of the approximative PEM and ML estimation are summarized in Table 2. The estimates of the nonlinear saturation function $f\big(x(t)\big)$ from Equation (58) are plotted in Figure 4.

Example 2

Par	True	Approx. PEM	ML
a_1	0.3000	0.3007 ± 0.2059	0.3091 ± 0.1735
a_2	-0.3000	-0.2805 ± 0.2193	-0.2922 ± 0.1846
b_1	-0.3000	-0.2889 ± 0.1600	-0.2932 ± 0.1339
b_2	0.3000	0.3031 ± 0.1109	0.3034 ± 0.0947
c_1	-0.4000	-0.2932 ± 0.0212	-0.4005 ± 0.0206
c_2	0.2000	0.0997 ± 0.0198	0.2004 ± 0.0205
λ_w	1.0000	n.e.	0.9734 ± 0.2020
λ_e	0.1000	n.e.	0.1000 ± 0.0074

Table 2
Parameter estimates with standard deviations for Example 2, using approximative PEM and ML. The mean and standard deviations are ncomputed over 1000 runs. The notation n.e. stands for "not estimated" as the noise variances are not estimated with the approximate PEM.

Both simulations confirm that while a straight-forward, approximative PEM gives biased estimates, the Maximum Likelihood method derived in this paper gives a consistent estimate of the system parameters, including noise variances, even when starting the numerical search in the biased estimate obtained from the approximative PEM. In these examples, also the variance of the approximative PEM estimates is larger than the estimates from the ML method.

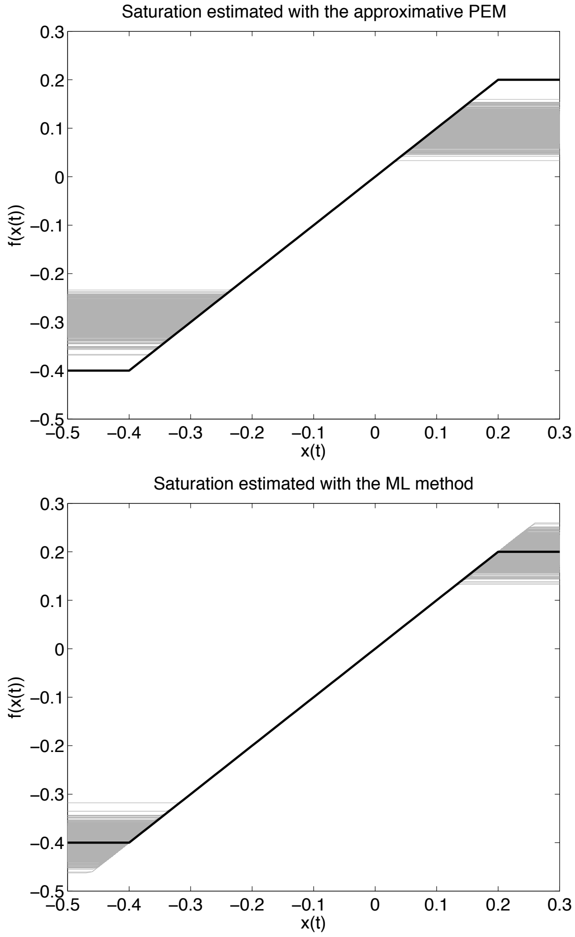


Fig. 4. Example 2: The true saturation curve as a thick black line and the 1000 estimated saturations, appearing as a grey zone. Above: approximative PEM. Below: ML.

5.3 Colored Process Noise

The process noise w entered at the ouput of the linear system in Figure 1 is typically composed of disturbances entering in various stages in the linear system. It is therefore somewhat restrictive to assume it to be white, and it is essential to test the algorithm for colored process noise.

We tested the case with

$$w(t) = \frac{0.3q^{-1}}{1 - 0.7q^{-1}}\bar{w}(t) \tag{59}$$

where $\bar{w}(t)$ white Gaussian noise with variance 4 (Example 1) and variance 1 (Example 2). (The variances of w will be 0.7059 and 0.1765 in the two cases.)

As detailed in Section 3.3, the ML criterion in this case includes filtered versions of the integration variable x, which makes the integral multidimensional. Instead of evaluating this integral, we use the simplified (but in this case, false) assumption that the noise is white, and thus apply the same algorithm as in the previous section.

The results are summarized in Tables 3 and 4.

As seem from these two tables, the ML method still produced consistent estimates of the system parameters. The estimated noise covariance corresponds to the covariance of the noise added to the output of the linear system (w).

Par	True	Approx. PEM	ML
c_0	0	0.7052 ± 0.1183	0.0063 ± 0.0872
c_1	1.0000	1.0020 ± 0.1553	1.0029 ± 0.1377
c_2	1.0000	1.0004 ± 0.0815	1.0017 ± 0.0622
a_0	0.5000	0.5003 ± 0.0286	0.4996 ± 0.0215
λ_w	0.7059	n.e.	0.7005 ± 0.0797
λ_e	1.0000	n.e.	1.0019 ± 0.0972

Table 3
Means and variances of the parameter estimates in Example 1, but using colored process noise.

Par	True	Approx. PEM	ML
a_1	0.3000	0.2873 ± 0.1181	0.2980 ± 0.1000
a_2	-0.3000	-0.2852 ± 0.1224	-0.2980 ± 0.1058
b_1	-0.3000	-0.3005 ± 0.0886	-0.3009 ± 0.0752
b_2	0.3000	0.3046 ± 0.0672	0.3025 ± 0.0560
c_1	-0.4000	-0.3686 ± 0.0198	-0.4011 ± 0.0191
c_2	0.2000	0.1712 ± 0.0171	0.2011 ± 0.0166
λ_w	0.1765	n.e.	0.1724 ± 0.0540
λ_e	0.1000	n.e.	0.0995 ± 0.0057

Table 4
Parameter estimates with standard deviations for Example 2 with colored noise, using approximative PEM and ML. See Table 2 for details.

One may discuss if this consistency for colored process noise is surprising or not. It is well known from linear identification that the Output Error approach gives constistent estimates, even when the output error disturbance is colored, and thus an erroneous likelihood criteron is used, Ljung (1999). The Wiener model resembles the output error model in that, in essence, it is a static model: For given input u noise is added to the deterministic variable $\eta(t) = G(q)u(t)$ as $\eta(t)+e(t)$ (linear output error) or as $f(\eta(t)+w(t))+e(t)$ (Wiener model). The spectrum or time correlation of the noises do not seem essential. However, a formal proof of this does not appear to be straightforward in the Wiener case.

6 Summary and Conclusions

In the quite extensive literature on Wiener model estimation, the most studied method has been to minimize the criterion (2). We have called that approach the

Approximative Prediction Error Method in this contribution. This method apparently is also the dominating approach for Wiener models in available software packages, like Ljung (2007) and Ninness & Wills (2006). We have in this contribution shown that this approach may lead to biased estimates in common situations. If disturbances are present in the system before the nonlinearity at the output, the estimates of the linear part and the nonlinearity will typically be biased, even when true descriptions are available in the model parameterizations. For example, Figure 4 clearly shows the bias in the estimate of an output saturation, in an otherwise ideal situation: Gaussian input, unbiased estimate of the linear part of the model. The reason for the bias is, in short, that the disturbances when transformed to the output error are no longer zero mean and independent of the input.

These deficiencies of the Approximative Prediction Error Method led us to a more serious statistical study of the Wiener model problem in the realistic case of both disturbances at the output measurements and process disturbances inside the dynamic part. We formulated the Likelihood function for the full problem. Although the maximization of this function at first sight may appear forbidding, an algorithm was developed that is not considerably more time-consuming than the Approximative Prediction Error Method. This ML method has the general property of consistency, which was also illustrated in the simulations.

In the general case of colored process noise, the Likelihood function is more complex to evaluate. However, in tests it has been found that the ML-method based on an assumption of white process noise produce consistent results also in the colored noise case. No proof of this observation has been established, though. A further challenge is to find efficient methods to evaluate the true likelihood function for this situation.

References

Bai, E.-W. (2003), 'Frequency domain identification of Wiener models', *Automatica* **39**(9), 1521–1530.

Billings, S. A. (1980), 'Identification of non-linear systems – a survey', *IEE Proc. D* **127**, 272–285.

Boyd, S. & Chua, L. O. (1985), 'Fading memory and the problem of approximating nonlinear operators with Volterra series', *IEEE Transactions on Circuits and Systems* **CAS-32**(11), 1150–1161.

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

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

Hagenblad, A. & Ljung, L. (2000), Maximum likelihood estimation of wiener models, *in* 'Proc. 39:th IEEE Conf. on Decision and Control', Sydney, Australia, pp. 2417–2418.

Hsu, K., Vincent, T. & Poolla, K. (2006), A kernel based approach to structured nonlinear system identification part i: Algorithms, part ii: Convergence and consistency, *in* 'Proc. IFAC Symposium on System Identification', Newcastle.

Hunter, I. W. & 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. & 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.

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

Ljung, L. (2001), 'Estimating linear time invariant models of non-linear time-varying systems', *European Journal of Control* **7**(2-3), 203–219. Semi-plenary presentation at the European Control Conference, Sept 2001.

Ljung, L. (2007), *The System Identification Toolbox: The Manual*, The MathWorks Inc. 1st edition 1986, 7th edition 2007, Natick, MA, USA.

Ninness, B. & Wills, A. (2006), An identification toolbox for profiling novel techniques, *in* '16th IFAC symposium on system identification'. `http://sigpromu.org/idtoolbox/`.

Nocedal, J. & Wright, S. J. (2006), *Numerical Optimization, Second Edition*, Springer-Verlag, New York.

Press, W. H., Teukolsky, S. A., Vetterling, W. A. & Fannery, B. P. (1992), *Numerical Recipes in C, the Art of Scientific Computing, Second Edition*, Cambridge University Press, Cambridge.

Schoukens, J., Nemeth, J. G., Crama, P., Rolain, Y. & Pintelon, R. (2003), 'Fast approximate identification of nonlinear systems', *Automatica* **39**(7), 1267–1274. July.

Van Overschee, P. & DeMoor, B. (1996), *Subspace Identification of Linear Systems: Theory, Implementation, Applications*, Kluwer Academic Publishers.

Westwick, D. & 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.

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

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