Technical report from Automatic Control at Linköpings universitet

Grey-box identification based on horizon estimation and nonlinear optimization

Alf Isaksson, David Törnqvist, Johan Sjöberg, Lennart Ljung

Division of Automatic Control

E-mail: alf@isy.liu.se, tornqvist@isy.liu.se, johans@isy.liu.se, ljung@isy.liu.se

28th June 2010

Report no.: LiTH-ISY-R-2963

Submitted to Proceedings of the 41st ISCIE International Symposium on Stochastic Systems theory and its applications (SSS'09), Konan University, Kobe, Japan, 13-14 November, 2009, pp 1-6.

Address:
Department of Electrical Engineering
Linköpings universitet
SE-581 83 Linköping, Sweden

WWW: http://www.control.isy.liu.se

AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

Abstract

In applications of (nonlinear) model predictive control a more and more common approach for the state estimation is to use moving horizon estimation, which employs (nonlinear) optimization directly on a model for a whole batch of data. This paper shows that horizon estimation may also be used for joint parameter estimation and state estimation, as long as a bias correction based on the Kalman filter is included. A procedure how to approximate the bias correction for nonlinear systems is outlined.

Keywords:

Grey-box identification based on horizon estimation and nonlinear optimization

Alf J. Isaksson*†
E-mail: alf.isaksson@se.abb.com
David Törnqvist†
Johan Sjöberg*†
Lennart Ljung†
*ABB AB, Corporate Research, SE-721 78, Västerås, Sweden
†Dept. Electrical Eng., Linköping University, SE-581 83 Linköping, Sweden

Abstract

In applications of (nonlinear) model predictive control a more and more common approach for the state estimation is to use moving horizon estimation, which employs (nonlinear) optimization directly on a model for a whole batch of data. This paper shows that horizon estimation may also be used for joint parameter estimation and state estimation, as long as a bias correction based on the Kalman filter is included. A procedure how to approximate the bias correction for nonlinear systems is outlined.

1 Introduction

This paper ultimately deals with parameter estimation in non-linear models. What triggers our interest in this area is the modelling for nonlinear model predictive control. While linear model predictive control has long been an established industrial area, nonlinear MPC has only found industrial applications more recently, see for example [5] and [9].

To obtain models for real industrial plants, there are two main alternatives available: Deriving a model from first principles using laws of physics, chemistry, etc, so-called white-box modelling, or estimating an empirical model from experimental data – black-box modelling. In general, a white-box model becomes a set of nonlinear Differential and Algebraic Equations (DAE). A black-box model on the other hand is typically linear, although non-linear black-box structures do exist (e.g. neural nets), and are usually estimated in a stochastic framework.

This paper will focus on the combination of these two techniques, i.e. to start with a physical model, but adding black-box elements to this model. This is often referred to as grey-box modelling or grey-box identification. The traditional solution for parameter estimation in grey-box modelling is to use a maximum likelihood criterion formed by running an extended Kalman filter. It will be argued that an interesting alternative is to work directly with a discretized version of the nonlinear model and fit both states and parameters using nonlinear optimization.

2 Model predictive control

As the name model predictive control indicates a crucial element of an MPC application is the model on which the control is based. Therefore, before a controller can be implemented a model has to be established. There are two main alternatives available for obtaining the model

- Deriving a model from first principles using laws of physics, chemistry, etc, so-called white-box modelling
- Estimating an empirical model from experimental data, black-box modelling

In general, a white-box model becomes a DAE

$$\begin{aligned} 0 &= f(\dot{x}(t), x(t), u(t)) \\ y(t) &= h(x(t), u(t)) \end{aligned}$$

where $y(t)$ denotes the measured process variables, and $u(t)$ the manipulated variable, i.e. the output of the MPC. Finally the internal variable $x(t)$ is what is usually referred to as the state of the system.

A black-box model on the other hand, is typically linear, but most often also discrete in time

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k \end{aligned}$$

Here the integer k denotes the k:th time index for which the signal value is available, i.e. at time kT_s, where T_s is the sampling interval. Hence, we have for example $x_k = x(kT_s)$.

The core of MPC is optimization. In each iteration of the control, i.e. any time a new measurement is collected, two optimization problems have to be solved (both using the model as an equality constraint); one using past data to estimate the current state vector $x(t)$ and one to optimize the future control variables. When solving the forward optimization problem a number of future values of the manipulated variables are calculated. However, only the values at the first time instant are transmitted to the underlying process. At the next time instant the optimizations are repeated, with the

optimization windows shifted one time step. This is known as receding horizon control, and is in fact what makes this a feedback control method. Performing optimization just once would correspond to open-loop control. The emphasis of this paper is on the second step – the state estimation – which will be presented in some more detail in the next subsection.

2.1 State Estimation

For the state estimation the optimization target is to obtain the best estimate of the internal variable x using knowledge of y and u, to be used as starting point for the forward optimization. This can be done using a Kalman filter (for an old classic see [1]) – or if the model is nonlinear an extended Kalman filter – where a stochastic modeling of process and measurement noises is applied. A Kalman filter is a recursive method, meaning that it takes only the most recent values of y_k and u_k to update the previous estimate $\hat{x}_{k-1}$ to obtain the new one $\hat{x}_k$. Hence it does not actually solve an optimization problem on-line. Kalman filtering is done in a statistical framework by adding process noise and measurement noise to the discrete-time state space system given in the previous section

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + w_k \\ y_k &= Cx_k + v_k \end{aligned}$$

where w_k and v_k are white Gaussian noises with covariance matrices Q and R respectively. Since we want to use certain quantities in the calculation later, the complete set of Kalman filter equations is given below:

$$\begin{aligned} S_k &= CP_{k|k-1}C^T + R && (1) \\ K_k &= P_{k|k-1}C^T S_k^{-1} && (2) \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k(y_k - C\hat{x}_{k|k-1}) \\ P_{k|k} &= (I - K_k C)P_{k|k-1} \\ \hat{x}_{k+1|k} &= A\hat{x}_{k|k} + Bu_k \\ P_{k+1|k} &= AP_{k|k}A^T + Q \end{aligned}$$

With access to more computational power, a much newer and increasingly popular approach is to use so-called moving horizon estimation (MHE) [10] instead. Then the process and measurement noise introduced are used as slack variables in the optimization. If the model is non-linear these slack variables are usually introduced in a discretized version of the model

$$\begin{aligned} x_{k+1} &= g(x_k, u_k) + w_k \\ y_k &= h(x_k, u_k) + v_k \end{aligned}$$

Moving horizon estimation then corresponds to minimizing

$$\begin{aligned} V &= (x_{k-M+1} - \hat{x}_{k-M+1})^T P^{-1} (x_{k-M+1} - \hat{x}_{k-M+1}) \\ &+ \sum_{n=k-M+1}^{k} w_n^T Q^{-1} w_n + v_n^T R^{-1} v_n \qquad (3) \end{aligned}$$

with respect to all states x_n within the horizon and possibly subject to constraints as, for example,

$$x_{\min} \leq x_n \leq x_{\max}$$

Here P, Q and R are weight matrices used for tuning of the estimator, which have a similar interpretation and importance as the estimate and noise covariance matrices in Kalman filtering.

As indicated in its name, the optimization for moving horizon estimation is typically done over a horizon of data $[t-(M-1)T_s, t]$, where t is the current measurement time. Since this time interval is in the past, we assume access to historic values of the applied manipulated variables u_k and the measured process variables y_k. The first penalty term in the criterion is called the arrival cost. It is to create a link from one optimization window to the next, where $\hat{x}_{k-M+1}$ denotes the estimate for this particular time instant from the optimization run at the previous cycle. Figure 1 tries to illustrate the MHE optimization which is a weighted sum of the the vertical bars in the upper and lower plots.

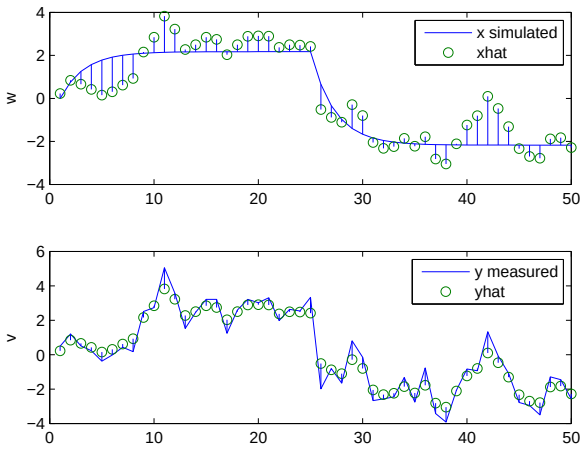

Figure 1: Illustration of moving horizon estimation.

Notice that in the optimization problems described above, the model (nonlinear or linear) should be considered as an equality constraint. We will not go into how these optimization problems are solved. Let us just point out that depending on the objective function and constraints (most importantly the model) different type of optimization problems result, leading to different type of optimization solvers needed. For example, a quadratic objective together with linear constraints corresponds to a quadratic programming (QP) problem, whereas a nonlinear objective or nonlinear model yields a nonlinear programming (NLP) problem which of course is much more difficult to solve. The latter case, when combined with the forward optimization, is usually referred to as a nonlinear MPC problem (NMPC).

2.2 Kalman filter vs MHE

Moving horizon estimation is identical to Kalman filtering if (see [10])

- the model is linear and discrete, i.e. given by matrices (A, B, C)
- the horizon $M = 1$
- the arrival cost is given by the Kalman filter Riccati equation as $P_{k-M+1|k-M}$
- the noises are zero-mean Gaussian with $Ev_k v_k^T = R$ and $Ew_k w_k^T = Q$
- there are no active constraints

When $M > 1$ the MHE estimates are identical to fixed interval smoothing estimates (see e.g. [1]) if the arrival cost is given by the corresponding forward-backward Riccati equation as $P_{k-M+1|k-1}$. The big difference between the two methods is that the MHE optimization is easily extended with constraints.

For nonlinear systems this exact similarity does of course no longer hold, why extended Kalman filtering and MHE will produce different results. Again constraint handling as well as the fact that the model only needs to be fulfilled at convergence (particularly useful for unstable processes) are advantages for MHE, while the heavier computational burden is a disadvantage.

3 Parameter estimation

Even if physical modelling is used there are often parameters whose numerical values are unknown, and therefore need to be estimated from data. Furthermore it is typically not easy to know at what level of detail to perform the physical modelling. A systematic procedure to gradually build up non-linear models using collected process data is usually referred to as grey-box identification. One such procedure is presented in the book by Bohlin [2], with which also follows a Matlab based software called MoCaVa. The software was also presented in [3], and contains modules for

- Preparation of data
- Simulation – To solve the (nonlinear) differential equations for fixed parameters and known inputs
- Calibration – to start with a root model and extend model gradually
- Validation – to test the model for a specific purpose

The part of the procedure of particular interest in this paper is the *Calibration* which includes the steps:

- Start with the root model
- For each extension, make a fit to data
- Compute the statistical risk of rejecting the previous model

The goal of the calibration can be formulated as: "Find the simplest model that cannot be falsified using the available data".

For the parameter fit, MoCaVa uses a maximum-likelihood criterion based on the extended Kalman filter

$$\min_{\theta} V_{ML} = \sum_{k=1}^{N} \left((y_k - \hat{y}_k)^T S_k^{-1} (y_k - \hat{y}_k) + \log\det(S_k) \right) \tag{4}$$

where S_k is given by (1). Then for the computation of the statistical risk the fact that V_{ML} is (asymptotically) chi-squared distributed is utilized.

Now instead, having an MPC implementation based on moving horizon estimation, a valid question is whether one cannot use the same optimization code for the parameter estimation? Because the identification is made on a collected set of input and output data, an off-line identification would rather correspond to Horizon Estimation, since no moving window will be applied. The answer is that an ML citerion may be formed using horizon estimation, but in addition to the criterion (3) it will contain a bias correction term which (somewhat surprisingly) is exactly the same one as in (4), i.e the ML-criterion becomes

$$\min_{x_k,\theta} V_{ML} = \sum_{k=1}^{N} \left(v_k^T R^{-1} v_k + w_k^T Q^{-1} w_k + \log\det(S_k) \right) \tag{5}$$

See Appendix A for a detailed derivation of this expression.

A common approach is to not include any process noise w_k when performing the parameter estimation (see e.g. [11]), and only use the first term in (5) with fixed weighting matrix. This is, in the system identification literature (e.g. [8]), known as Output Error (OE) and will also lead to unbiased estimates. Notice, however, that if we want to include the measurement covariance R as a free estimation parameter, then the bias correction term is indeed necessary, otherwise there is nothing preventing the covariance estimate from tending to infinity.

4 Monte-Carlo simulation

In this section we will illustrate the theoretical result of the previous section by simulating a simple first order (linear) process for many different noise realizations.

The chosen example is given by

$$\begin{aligned} x_{k+1} &= ax_k + bu_k + w_k \\ y_k &= x_k + v_k \end{aligned}$$

where both w_k and v_k are zero mean Gaussian noises with unit variance.

The particular example $a = 0.7$ and $b = 0.3$ was simulated for 100 different noise realizations (one such simulation is shown in Figure 2) and the optimal parameter estimates where found using three different criteria:

HE Horizon estimation is also the same as the maximum a posteriori estimate using a flat prior, i.e.

$$\begin{aligned} \min_{x_k,a,b} V_{HE} &= \sum_{k=1}^{N}(w_k^2 + v_k^2) \\ x_{k+1} &= ax_k + bu_k + w_k \\ y_k &= x_k + v_k \end{aligned}$$

ML Horizon estimation with the bias correction:

$$\begin{aligned} \min_{x_k,a,b} V_{ML} &= \sum_{k=1}^{N}(w_k^2 + v_k^2 + \log(s_k)) \\ x_{k+1} &= ax_k + bu_k + w_k \\ y_k &= x_k + v_k \\ p_{k+1} &= a^2 p_k^2/(p_k + 1) + 1 \\ s_k &= p_k + 1 \end{aligned}$$

OE Output error, i.e. estimating the parameters from a pure simulation model

$$\begin{aligned} \min_{a,b} V_{OE} &= \sum_{k=1}^{N}(y_k - \hat{y}_k) \\ \hat{y}_{k+1} &= a\hat{y}_k + bu_k \end{aligned}$$

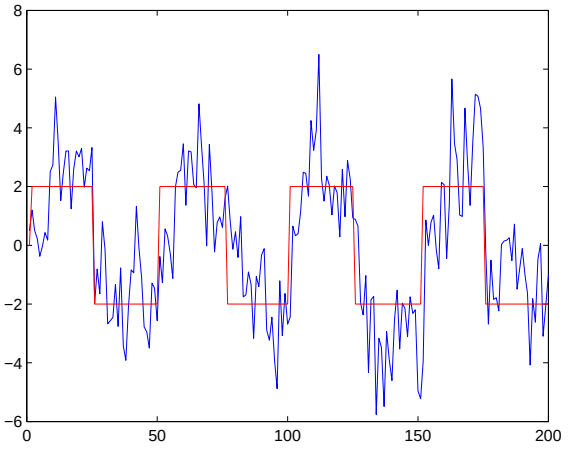

Figure 2: Example of one signal realization in the Monte-Carlo simulation

The parameter estimates are shown in Figure 3, where each cross corresponds to one estimate for one realization. From the figure it is clear that pure Horizon Estimation leads to significant bias in the parameter estimates, while both Maximum-Likelihood and Output Error are unbiased. However, as expected, ML clearly has much lower covariance of the parameter estimates.

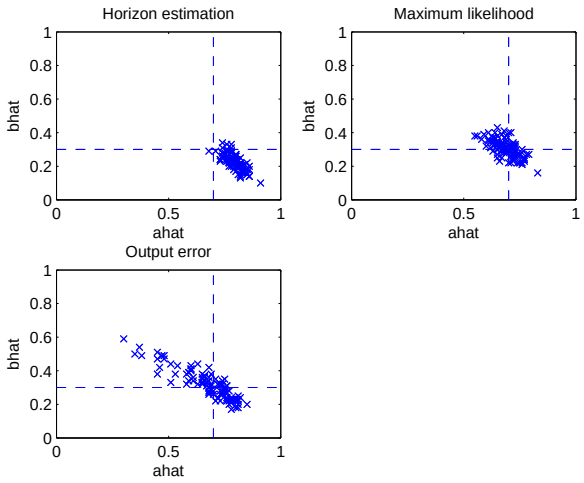

Figure 3: Monte-Carlo simulation comparing three different identification methods for first-order system.

5 Suggested procedure for non-linear models

While the derivation and simulations above are for linear systems, the end goal is of course to use this new optimization approach for identification of non-linear models. Similar to extended Kalman filtering we suggest to use the linearized model, but only for the computation of the bias correction term. The rest of the optimization deploys the full non-linear model (although discretized).

1. Model process, for example, in Modelica
2. Discretize symbolically and export equations
3. Linearize model symbolically
4. Import and prepare data
5. Carefully introduce noise variables at equations motivated by physical insight
6. Solve by nonlinear programming (e.g. using IPOPT)

$$\min_{x_k,\theta} V_{ML} = \sum_{k=1}^{N}\left(w_k^T Q^{-1} w_k + v_k^T R^{-1} v_k + \log\det(S_k)\right)$$

subject to

$$\begin{aligned} x_{k+1} &= g(x_k, u_k) + w_k \\ y_k &= h(x_k, u_k] + v_k \\ x_{\min} &\le x_k \le x_{\max} \\ \theta_{\min} &\le \theta \le \theta_{\max} \end{aligned}$$

7. For every evaluation of V calculate the (time varying) linearized system along the trajectory to compute S_k

8. Test which parameters to make free (including noise parameters) by hypothesis testing using the chi-squared risk calculation

9. Repeat 5-8 until no further improvement

6 Conclusions

The process modelling is by far the most time consuming part of an MPC project. When physical modelling is used a procedure for gradual extension of the model has been designed by Bohlin [2], and is called grey-box identification. This procedure is built around maximum-likelihood identification. In its original form the ML criterion is evaluated using (extended) Kalman filters.

In this paper we have advocated that if the MPC implementation uses moving horizon estimation for the state estimation, this optimization code may be used also for the grey-box identification, as long as one includes a bias correction term.

This is research still in progress and several issues are remaining such as

- How to deal with size and sparsity of P matrix
- The relationship between discretization of the DAE and the bias correction
- Creating a user friendly software environment similar to MoCaVa

Acknowledgement

The first author is very grateful for the invitation to deliver the Sunahara Keynote Lecture at SSS'09, which triggered this whole research effort.

A Derivation of parameter estimation using Horizon estimation

Parameter estimation with the Maximum Likelihood (ML) method gives unbiased parameter estimates. Using a flat prior for the parameters and computing the Maximum Aposteriori (MAP) estimate will give biased estimates. Here, it will be shown how to select the prior in order to get unbiased estimates.

The following densities are given as a model of the system: $p(y|x,\theta)$, $p(x|\theta)$, and $p(\theta)$. The states, x, and parameters, θ, should be estimated given a measurement, y. This could be done either with maximum-likelihood (ML) estimation as

$$\hat{\theta} = \arg\max_{\theta} p(y|\theta) = \int p(y|x,\theta)p(x|\theta)\,dx$$

or as maximum a posteriori (MAP) estimation

$$\hat{\theta} = \arg\max_{x,\theta} p(x,\theta|y) = \frac{p(y|x,\theta)p(x|\theta)p(\theta)}{p(y)}.$$

To analyze the difference between ML and MAP, rewrite the MAP estimation density as

$$p(x,\theta|y) \propto p(y|x,\theta)p(x|\theta)p(\theta) = \underbrace{p(y|\theta)}_{\text{ML density}}\ p(x|y,\theta)p(\theta), \tag{6}$$

were it can be seen that MAP adds a factor $p(x|y,\theta)p(\theta)$ to the ML density. If the MAP solution should equal ML, $p(\hat{x}^{\text{MAP}}|y,\theta)p(\theta)$ has to be independent of θ ($\hat{x}^{\text{MAP}} = \arg\max_x p(x|y,\theta)$). Selecting a $p(\theta)$ that fulfills this requirement corresponds to a noninformative prior.

Linear Gaussian System

The result above is here derived for the special case of a linear Gaussian system, where the data is given for a time window. Consider a linear state space model on the following form

$$\begin{aligned} x_{k+1} &= A_k(\theta)x_k + w_k, \\ y_k &= B_k(\theta)x_k + v_k, \end{aligned}$$

where the system matrices A and C are dependent on the parameter θ, x is the system state, y is the system output, w and v are zero-mean Gaussian process and measurement noise with covariances Q and R respectively. This system can be described in batched form over a time window, directly derived from the linear state-space form. In addition to moving horizon estimation, the batch form is often used in fault detection/diagnosis, see [4, 6]. Stack N signal values to define signal vectors like $Y = \left(y_1^T, \ldots, y_N^T\right)^T$, for all signals except the input noise vector which is defined as $W = \left(w_0^T, \ldots, w_{N-1}^T\right)^T$. If the initial state is zero (the framework can easily be extended to nonzero initial states), the signals may be described as

$$X = \bar{F}(\theta)W, \tag{7}$$

$$Y = \bar{H}(\theta)X + V, \tag{8}$$

where

$$\bar{F}(\theta) = \begin{pmatrix} I & 0 & 0 & \cdots & 0 \\ A_1 & I & 0 & \cdots & 0 \\ A_2A_1 & A_2 & I & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ \prod_{n=1}^{N-1} A_n & \cdots & & A_{N-1} & I \end{pmatrix}$$

and $\bar{H}(\theta) = \text{diag}(C_1, \cdots, C_N)$. The noise covariances are given by $\bar{Q} := \text{cov}W = \text{diag}(Q, \cdots, Q)$ and $\bar{R} := \text{cov}(V) = \text{diag}(R, \cdots, R)$.

To simplify notation, denote $\bar{F} = \bar{F}(\theta)$ and $\bar{H} = \bar{H}(\theta)$. Now, the equations (7) and (8) can be described as the following probability densities

$$\begin{aligned} p(X|\theta) &= N(X; 0, \bar{F}\bar{Q}\bar{F}^T), \\ p(Y|X,\theta) &= N(Y; \bar{H}X, \bar{R}). \end{aligned}$$

The likelihood density is computed as

$$p(Y|\theta) = \int p(Y|X,\theta)p(X|\theta)\,dX = N(Y; 0, \bar{S}),$$

where $\bar{S} = \bar{H}\bar{F}\bar{Q}\bar{F}^T\bar{H}^T + \bar{R}$. The calculations are analogous to those made for the Kalman filter time update [7]. The distribution for the states in the window is computed as

$$p(X|Y,\theta) = \frac{p(Y|X,\theta)p(X|\theta)}{p(Y|\theta)} = N(X; KY, \bar{P})$$

where $K = \bar{F}\bar{Q}\bar{F}^T\bar{H}^T\bar{S}^{-1}$ and $\bar{P} = ((\bar{F}\bar{Q}\bar{F}^T)^{-1} + \bar{H}^T\bar{R}^{-1}\bar{H})^{-1}$. For this result, the calculations are analogous to the Kalman filter measurement update. For its maximum $\hat{X}^{\mathrm{MAP}} = KY$, the exponent is zero and the probability function evaluates to

$$p(\hat{X}^{\mathrm{MAP}}|Y,\theta) = \frac{1}{(2\pi)^{n/2}\det{}^{1/2}\bar{P}}.$$

which is clearly dependent of θ since $\bar{P}$ is a function of θ. In order to fulfill that $p(\hat{X}^{\mathrm{MAP}}|Y,\theta)p(\theta)$ is independent of θ (condition from (6)), the prior must be selected as

$$p(\theta) = \det{}^{1/2}\bar{P},$$

which is the noninformative prior.

The cost function to minimize for the MAP estimate is then given by

$$\begin{aligned} - &\quad \log p(X,\theta|Y) \propto -\log p(Y|\theta)p(X|Y,\theta)p(\theta) \\ = &\quad \frac{1}{2}Y^T\bar{S}^{-1}Y + (X-KY)^T\bar{P}^{-1}(X-KY) \\ + &\quad \frac{1}{2}\log\det\bar{S} + \frac{1}{2}\log\det\bar{P} - \frac{1}{2}\log\det\bar{P}, \end{aligned}$$

where the prior cancels the covariance term of $p(X|Y,\theta)$. The MAP estimate of X is $\hat{X}^{\mathrm{MAP}} = KY$ which will make the X-dependent term zero in optimum, the remaining terms are the same as for ML estimation. Another way of writing this cost function is as a function of the basic densities

$$-\log p(X,\theta|Y) \propto -\log p(Y|X,\theta)p(X|\theta)p(\theta) \quad (9)$$

$$= \frac{1}{2}(Y-\bar{H}X)^T\bar{R}^{-1}(Y-\bar{H}X) + \frac{1}{2}X^T(\bar{F}\bar{Q}\bar{F}^T)^{-1}X \quad (10)$$

$$+\frac{1}{2}\log\det\bar{R} + \frac{1}{2}\log\det\bar{F}\bar{Q}\bar{F}^T - \frac{1}{2}\log\det\bar{P} \quad (11)$$

Notice that the two terms in (10) correspond exactly to the moving horizon estimation criterion (3). For the three remaining terms in (11) observe that

$$\log\det\bar{R} + \log\det\bar{F}\bar{Q}\bar{F}^T - \log\det\bar{P} =$$

$$\log\det\bar{R}\det(I + \bar{F}\bar{Q}\bar{F}^T\bar{H}^T\bar{R}^{-1}\bar{H}) = \frac{1}{2}\log\det\bar{S}$$

where the last equality follows from the determinant identity

$$\det(I+AB) = \det(I+BA)$$

To compute $\bar{S} = EYY^T$ we apply the innovations form for the linear stochastic state space model (see e.g. [1]):

$$\begin{aligned} \bar{x}_{k+1} &= A_k(\theta)\bar{x}_k + K_k\nu_k, \\ y_k &= B_k(\theta)\bar{x}_k + \nu_k, \end{aligned}$$

where K_k is the Kalman filter gain given by (2) and ν_k is the prediction error which has covariance S_k given by (1). Utilizing a batch form similar to (7) and (8) we get

$$\log\det\bar{S} = \sum_{k=1}^{N}\log\det(S_k)$$

which concludes the derivation of (5).

References

[1] B.D.O. Anderson and J.B. Moore. *Optimal Filtering.* Prentice-Hall, 1979.

[2] T. Bohlin. *Practical Grey-box Process Identification – Theory and Applications.* Springer, 2006.

[3] T. Bohlin and A.J. Isaksson. Grey-Box Model Calibrator & Validator, Proceedings Springer, 2006, *Proceedings IFAC SYSID*, Rotterdam, The Netherlands, 2003.

[4] A.Y. Chow and A.S. Willsky, Analytical Redundancy and the Design of Robust Failure Detection Systems,*IEEE Trans. Autom. Control*, Vol. 29, No. 7, pp. 603–614, 1984.

[5] R. Franke and L. Vogelbacher, Nonlinear model predictive control for cost optimal startup of steam power plants, *at – Automatisierungstechnik*, Vol 54, No. 12, 2006.

[6] F. Gustafsson, *Adaptive filtering and change detection*, John Wiley & Sons, Ltd, 2nd Edition, 2001.

[7] Y.C. Ho and R.C.K. Lee, A Bayesian Approach to Problems in Stochastic Estimation and Control, *IEEE Trans. Autom. Control*, Vol. 9, pp. 333–338, 1964.

[8] L. Ljung, *System Identification – Theory for the User*, 2nd edition, Prentice-Hall, Upper Saddle River, N.J., 1999.

[9] J. Pettersson, L. Ledung and X. Zhang, Decision support for pulp mill operations based on large-scale on-line optimization, *Preprints of Control Systems 2006*, Tampere 6-8 June, 2006.

[10] C.V. Rao, J.B. Rawlings and J.H. Lee. Constrained linear state estimation – a moving horizon approach, *Automatica*, 37(10):1619-1628, 2001.

[11] V.M. Zavala, *Computational Strategies for the Operation of Large-Scale Chemical Processes*, PhD Thesis, Carnegie Mellon University, 2008.

Avdelning, Institution
Division, Department

Division of Automatic Control
Department of Electrical Engineering

Datum
Date

2010-06-28

Språk
Language

☐ Svenska/Swedish
☒ Engelska/English

☐ _____

Rapporttyp
Report category

☐ Licentiatavhandling
☐ Examensarbete
☐ C-uppsats
☐ D-uppsats
☒ Övrig rapport
☐ _____

ISBN
—

ISRN
—

Serietitel och serienummer
Title of series, numbering

ISSN
1400-3902

LiTH-ISY-R-2963

URL för elektronisk version

http://www.control.isy.liu.se

Titel
Title

Grey-box identification based on horizon estimation and nonlinear optimization

Författare
Author

Alf Isaksson, David Törnqvist, Johan Sjöberg, Lennart Ljung

Sammanfattning
Abstract

In applications of (nonlinear) model predictive control a more and more common approach for the state estimation is to use moving horizon estimation, which employs (nonlinear) optimization directly on a model for a whole batch of data. This paper shows that horizon estimation may also be used for joint parameter estimation and state estimation, as long as a bias correction based on the Kalman filter is included. A procedure how to approximate the bias correction for nonlinear systems is outlined.

Nyckelord
Keywords