Available online at www.sciencedirect.com
SCIENCE DIRECT®

automatica

ELSEVIER

Automatica 40 (2004) 225–237

www.elsevier.com/locate/automatica

Parameter estimation in stochastic grey-box models[☆]

Niels Rode Kristensen[a], Henrik Madsen[b,*], Sten Bay Jørgensen[a]

[a] Department of Chemical Engineering, Technical University of Denmark, Building 229, DK-2800 Lyngby, Denmark
[b] Informatics and Mathematical Modelling, Technical University of Denmark, Building 321, DK-2800 Lyngby, Denmark

Received 20 November 2002; received in revised form 28 April 2003; accepted 25 September 2003

Abstract

An efficient and flexible parameter estimation scheme for grey-box models in the sense of discretely, partially observed Itô stochastic differential equations with measurement noise is presented along with a corresponding software implementation. The estimation scheme is based on the extended Kalman filter and features *maximum likelihood* as well as *maximum a posteriori* estimation on multiple independent data sets, including irregularly sampled data sets and data sets with occasional outliers and missing observations. The software implementation is compared to an existing software tool and proves to have better performance both in terms of quality of estimates for nonlinear systems with significant diffusion and in terms of reproducibility. In particular, the new tool provides more accurate and more consistent estimates of the parameters of the diffusion term.

Keywords: Grey-box models; Parameter estimation; Stochastic differential equations; Maximum likelihood estimation; Extended Kalman filter; Estimation with missing observations; Robust estimation; Estimation accuracy; Software tools

1. Introduction

The development of various methods for advanced model-based control (Bitmead, Gevers, & Wertz, 1990; Allgöwer & Zheng, 2000) and recent advances in sensor technology allowing these methods to be applied to an increasing number of complex physical, chemical and biological systems has rendered the development of high-quality models for such systems very important. In particular, since a model must be able to predict the future evolution of the system, it must capture the inherently nonlinear behaviour of such systems and it must provide means to accommodate noise in the form of process noise due to approximation errors or unmodelled inputs and measurement noise due to imperfect measurements.

White box models, derived from first principles, are often able to satisfy the former requirement but fail to satisfy the latter, whereas *black box* models, developed with methods for system identification (Ljung, 1987; Söderström & Stoica, 1989), satisfy the latter but often fail to satisfy the former. Stochastic state space models or *grey-box* models, which consist of a set of stochastic differential equations (SDEs) describing the dynamics of the system in continuous time and a set of discrete time measurement equations, provide a way of combining the advantages of both model types by allowing prior physical knowledge to be incorporated and statistical methods for parameter estimation to be applied. Bohlin and Graebe (1995) even argue that such models provide a natural framework for modelling dynamic systems.

Apart from the work by Bohlin and Graebe (1995) and earlier work by the authors of the present paper, mathematical modelling of dynamic systems based on SDEs has received limited attention in the control and system identification communities since Jazwinski (1970) and Åström (1970). This is evident from a series of review papers on identification of continuous time models (Young, 1981; Unbehauen & Rao, 1990, 1998; Nielsen, Madsen, & Young, 2000). Due to the potential benefits of grey-box models, utilized by, e.g. Madsen and Holst (1995) and Jacobsen and Madsen (1996), it is the opinion of the authors that the topic deserves much more attention.

Particular benefits of grey-box models as opposed to black box models include the fact that physical knowledge and other prior information can be incorporated directly. This typically yields models with fewer and physically

[☆] This paper was not presented at any IFAC meeting. This paper was recommended for publication in revised form by Associate Editor Johan Schoukens under the direction of Editor Torsten Söderström.

[*] Corresponding author. Tel.: +45-45253408; fax: +45-45882673.
E-mail address: hm@imm.dtu.dk (H. Madsen).

0005-1098/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2003.10.001

meaningful parameters, which are valid over much wider ranges of state space. As opposed to white box models parameter estimation in grey-box models tends to give more reproducible results and less bias, because random effects due to process and measurement noise are not absorbed into the parameter estimates but specifically accounted for by the diffusion and measurement noise terms. Furthermore, simultaneous estimation of the parameters of these terms provides an estimate of the uncertainty of the model, upon which further model development can be based. In particular, estimates of the parameters of the diffusion term can be used to assess the quality of a model (Kristensen, Madsen, & Jørgensen, 2001), to discriminate between different models (Kristensen, Madsen, & Jørgensen, 2002), and to pinpoint model deficiencies and subsequently uncover their structural origin (Kristensen, Madsen, & Jørgensen, 2004). Thus, obtaining accurate and consistent estimates of the parameters of the diffusion term is very important.

The focus of the present paper is on estimation of unknown parameters in grey-box models, and the primary aim of the paper is to present an efficient and flexible scheme for performing the estimation and a software implementation of this scheme. Not all of the material is new, as some of the solutions presented have been implemented before. Indeed, a similar parameter estimation scheme and software tool has been presented by Bohlin and Graebe (1995). There are, however, a number of very important differences between the two schemes, and a secondary aim of the paper is to outline these differences and illustrate how they influence estimation performance. A key result is that the new tool provides more accurate estimates for nonlinear systems with significant diffusion and more consistent estimates, particularly with respect to the parameters of the diffusion term. The paper is organized as follows: The mathematical basis of the estimation scheme is presented in Section 2 and the software implementation is described in Section 3. The differences between the scheme presented here and the one by Bohlin and Graebe (1995) are outlined in Section 4, where the influence on estimation performance is also illustrated. The results are discussed in Section 5 and the conclusions are given in Section 6.

2. Mathematical basis

This section contains a condensed outline of the mathematics behind the proposed parameter estimation scheme and of the algorithms of the corresponding software implementation (see Section 3). A complete outline can be found in Kristensen and Madsen (2003).

2.1. General model structure

Adapting the terminology of Bohlin and Graebe (1995), the term *grey-box model* will be used throughout this paper for a model consisting of a set of nonlinear, discretely, partially observed SDEs with measurement noise, i.e.

$$\mathrm{d}\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})\,\mathrm{d}t + \boldsymbol{\sigma}(\boldsymbol{u}_t, t, \boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\omega}_t, \tag{1}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k, \tag{2}$$

where $t \in \mathbb{R}$ is the time variable (t_k, $k = 0, \dots, N$ are sampling instants); $\boldsymbol{x}_t \in \mathscr{X} \subset \mathbb{R}^n$ is a vector of state variables; $\boldsymbol{u}_t \in \mathscr{U} \subset \mathbb{R}^m$ is a vector of input variables; $\boldsymbol{y}_k \in \mathscr{Y} \subset \mathbb{R}^l$ is a vector of output variables; $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ is a vector of (possibly unknown) parameters; $\boldsymbol{f}(\cdot) \in \mathbb{R}^n$, $\boldsymbol{\sigma}(\cdot) \in \mathbb{R}^{n \times n}$ and $\boldsymbol{h}(\cdot) \in \mathbb{R}^l$ are nonlinear functions; $\{\boldsymbol{\omega}_t\}$ is an n-dimensional standard Wiener process and $\{\boldsymbol{e}_k\}$ is an l-dimensional white noise process with $\boldsymbol{e}_k \in \mathscr{N}(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{u}_k, t_k, \boldsymbol{\theta}))$. The first term on the right-hand side of (1) is commonly called the *drift* term and the second term is commonly called the *diffusion* term.

Remark 1. SDEs may be interpreted both in the sense of Stratonovich and in the sense of Itô, but since the Stratonovich interpretation is less suitable for parameter estimation (Jazwinski, 1970; Åström, 1970; Kloeden & Platen, 1992), the Itô interpretation is adapted here.

Remark 2. The diffusion term is assumed to be independent of the state variables, because this renders parameter estimation more feasible. However, as shown by Nielsen and Madsen (2001), a transformation may be applied for a restricted class of systems with such dependencies or *level effects*, allowing application of the proposed estimation scheme to such systems as well.

2.2. Parameter estimation methods

2.2.1. Maximum likelihood estimation

Given the model structure in (1) and (2) *maximum likelihood* (ML) estimates of the unknown parameters can be determined by finding the parameters $\boldsymbol{\theta}$ that maximize the likelihood function of a given sequence of measurements $\boldsymbol{y}_0, \boldsymbol{y}_1, \dots, \boldsymbol{y}_k, \dots, \boldsymbol{y}_N$. Introducing the notation

$$\mathscr{Y}_k = [\boldsymbol{y}_k, \boldsymbol{y}_{k-1}, \dots, \boldsymbol{y}_1, \boldsymbol{y}_0] \tag{3}$$

the likelihood function is the joint probability density

$$L(\boldsymbol{\theta}; \mathscr{Y}_N) = p(\mathscr{Y}_N | \boldsymbol{\theta}) \tag{4}$$

or equivalently

$$L(\boldsymbol{\theta}; \mathscr{Y}_N) = \left(\prod_{k=1}^{N} p(\boldsymbol{y}_k | \mathscr{Y}_{k-1}, \boldsymbol{\theta}) \right) p(\boldsymbol{y}_0 | \boldsymbol{\theta}), \tag{5}$$

where the rule $P(A \cap B) = P(A|B)P(B)$ has been applied to form a product of conditional densities.

In order to obtain an exact evaluation of the likelihood function, a general nonlinear filtering problem must be solved. Thus, the initial probability density function must be known and all subsequent conditional densities must be determined by successively solving Kolmogorov's forward equation and applying Bayes' rule (Jazwinski, 1970). In practice, this approach is computationally infeasible, however, and an alternative is needed. Nielsen et al. (2000) have recently reviewed the state of the art with respect to parameter estimation in discretely observed Itô stochastic differential equations. In the general case of higher order, partially observed systems with measurement noise they conclude that only methods based on approximate nonlinear filters provide a computationally feasible solution to the problem. However, since the diffusion term in (1) has been assumed to be independent of the state variables, a simpler alternative can be used. Since the SDEs in (1) are driven by a Wiener process, and since increments of a Wiener process are Gaussian, it is reasonable to assume, under some regularity conditions, that the conditional densities can be well approximated by Gaussian densities, which means that a method based on the extended Kalman filter (EKF), which is linear, can be applied. The assumption can (and should) be checked subsequent to the estimation (Holst, Holst, Madsen, & Melgaard, 1992; Bak, Madsen, & Nielsen, 1999).

The Gaussian density is completely characterized by its mean and covariance, so introducing the notation

$$\hat{\boldsymbol{y}}_{k|k-1} = E\{\boldsymbol{y}_k|\mathscr{Y}_{k-1},\boldsymbol{\theta}\}, \tag{6}$$

$$\boldsymbol{R}_{k|k-1} = V\{\boldsymbol{y}_k|\mathscr{Y}_{k-1},\boldsymbol{\theta}\} \tag{7}$$

and

$$\boldsymbol{\varepsilon}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1} \tag{8}$$

the likelihood function can be rewritten as

$$L(\boldsymbol{\theta};\mathscr{Y}_N) = \left(\prod_{k=1}^{N} \frac{\exp\left(-\frac{1}{2}\boldsymbol{\varepsilon}_k^{\mathrm{T}}\boldsymbol{R}_{k|k-1}^{-1}\boldsymbol{\varepsilon}_k\right)}{\sqrt{\det(\boldsymbol{R}_{k|k-1})}\left(\sqrt{2\pi}\right)^l}\right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \tag{9}$$

and the parameter estimates can be determined by conditioning on $\boldsymbol{y}_0$ and solving the optimization problem

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta}\{-\ln(L(\boldsymbol{\theta};\mathscr{Y}_N|\boldsymbol{y}_0))\}. \tag{10}$$

For each set of parameters $\boldsymbol{\theta}$ in the optimization, the innovations $\boldsymbol{\varepsilon}_k$ and their covariances $\boldsymbol{R}_{k|k-1}$ are computed recursively by means of the EKF, which consists of the output *prediction* equations

$$\hat{\boldsymbol{y}}_{k|k-1} = \boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1},\boldsymbol{u}_k,t_k,\boldsymbol{\theta}), \tag{11}$$

$$\boldsymbol{R}_{k|k-1} = \boldsymbol{C}\boldsymbol{P}_{k|k-1}\boldsymbol{C}^{\mathrm{T}} + \boldsymbol{S}, \tag{12}$$

the *innovation* equation

$$\boldsymbol{\varepsilon}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_{k|k-1}, \tag{13}$$

the Kalman *gain* equation

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{C}^{\mathrm{T}}\boldsymbol{R}_{k|k-1}^{-1}, \tag{14}$$

the *updating* equations

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k\boldsymbol{\varepsilon}_k, \tag{15}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{R}_{k|k-1}\boldsymbol{K}_k^{\mathrm{T}} \tag{16}$$

and the state *prediction* equations

$$\frac{\mathrm{d}\hat{\boldsymbol{x}}_{t|k}}{\mathrm{d}t} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{t|k},\boldsymbol{u}_t,t,\boldsymbol{\theta}), \tag{17}$$

$$\frac{\mathrm{d}\boldsymbol{P}_{t|k}}{\mathrm{d}t} = \boldsymbol{A}\boldsymbol{P}_{t|k} + \boldsymbol{P}_{t|k}\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{\sigma}\boldsymbol{\sigma}^{\mathrm{T}} \tag{18}$$

which are solved for $t\in[t_k,t_{k+1}[$. In the above equations the following notation has been applied:

$$\boldsymbol{A} = \left.\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{x}_t}\right|_{\hat{\boldsymbol{x}}_{k|k-1},\boldsymbol{u}_k,t_k}, \quad \boldsymbol{C} = \left.\frac{\partial\boldsymbol{h}}{\partial\boldsymbol{x}_t}\right|_{\hat{\boldsymbol{x}}_{k|k-1},\boldsymbol{u}_k,t_k},$$

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{u}_k,t_k,\boldsymbol{\theta}), \quad \boldsymbol{S} = \boldsymbol{S}(\boldsymbol{u}_k,t_k,\boldsymbol{\theta}).$$

Initial conditions for the EKF are $\hat{\boldsymbol{x}}_{t|t_0} = \boldsymbol{x}_0$, which can either be pre-specified or estimated along with the unknown parameters as a part of the overall problem, and $\boldsymbol{P}_{t|t_0} = \boldsymbol{P}_0$, which can be computed as follows:

$$\boldsymbol{P}_0 = P_s\int_{t_0}^{t_1} \mathrm{e}^{\boldsymbol{A}s}\boldsymbol{\sigma}\boldsymbol{\sigma}^{\mathrm{T}}(\mathrm{e}^{\boldsymbol{A}s})^{\mathrm{T}}\,\mathrm{d}s,$$

i.e. as the integral of the Wiener process and the system dynamics over the first sample, scaled by a pre-specified scaling factor $P_s \geqslant 1$.

The EKF is sensitive to nonlinear effects, and the approximate solution obtained by solving (17)–(18) may be too crude (Jazwinski, 1970). Moreover, the assumption of Gaussian conditional densities is only likely to hold for small sample times (and should therefore be checked subsequent to the estimation). To provide a better approximation, the time interval $[t_k,t_{k+1}[$ is subsampled, i.e. $[t_k,\ldots,t_j,\ldots,t_{k+1}[$, and the equations are linearized at each subsampling instant. This way the numerical solution of (17)–(18) can be simplified by applying the analytical solutions to the corresponding linearized propagation equations given by

$$\frac{\mathrm{d}\hat{\boldsymbol{x}}_{t|j}}{\mathrm{d}t} = \boldsymbol{f}_0 + \boldsymbol{A}(\hat{\boldsymbol{x}}_t - \hat{\boldsymbol{x}}_j) + \boldsymbol{B}(\boldsymbol{u}_t - \boldsymbol{u}_j), \tag{19}$$

$$\frac{\mathrm{d}\boldsymbol{P}_{t|j}}{\mathrm{d}t} = \boldsymbol{A}\boldsymbol{P}_{t|j} + \boldsymbol{P}_{t|j}\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{\sigma}\boldsymbol{\sigma}^{\mathrm{T}} \tag{20}$$

which are solved for $t\in[t_j,t_{j+1}[$. Here the notation

$$\boldsymbol{A} = \left.\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{x}_t}\right|_{\hat{\boldsymbol{x}}_{j|j-1},\boldsymbol{u}_j,t_j}, \quad \boldsymbol{B} = \left.\frac{\partial\boldsymbol{f}}{\partial\boldsymbol{u}_t}\right|_{\hat{\boldsymbol{x}}_{j|j-1},\boldsymbol{u}_j,t_j},$$

$$\boldsymbol{f}_0 = \boldsymbol{f}(\hat{\boldsymbol{x}}_{j|j-1},\boldsymbol{u}_j,t_j,\boldsymbol{\theta}), \quad \boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{u}_j,t_j,\boldsymbol{\theta})$$

has been applied, and the analytical solutions are

$$\begin{aligned}\hat{\boldsymbol{x}}_{j+1|j} = {} & \hat{\boldsymbol{x}}_{j|j} + \boldsymbol{A}^{-1}(\boldsymbol{\Phi}_s - \boldsymbol{I})\boldsymbol{f}_0 \\ & + (\boldsymbol{A}^{-1}(\boldsymbol{\Phi}_s - \boldsymbol{I}) - \boldsymbol{I}\tau_s)\boldsymbol{A}^{-1}\boldsymbol{B}\boldsymbol{\alpha},\end{aligned} \tag{21}$$

$$\boldsymbol{P}_{j+1|j} = \boldsymbol{\Phi}_s\boldsymbol{P}_{j|j}\boldsymbol{\Phi}_s^{\mathrm{T}} + \int_0^{\tau_s} \mathrm{e}^{\boldsymbol{A}s}\boldsymbol{\sigma}\boldsymbol{\sigma}^{\mathrm{T}}\mathrm{e}^{\boldsymbol{A}^{\mathrm{T}}s}\,\mathrm{d}s, \tag{22}$$

where $\tau_s = t_{j+1} - t_j$ and $\boldsymbol{\Phi}_s = e^{\boldsymbol{A}\tau_s}$, and where

$$\boldsymbol{\alpha} = \frac{\boldsymbol{u}_{j+1} - \boldsymbol{u}_j}{t_{j+1} - t_j} \tag{23}$$

has been introduced to allow assumption of either *zero-order hold* ($\boldsymbol{\alpha} = \boldsymbol{0}$) or *first-order hold* ($\boldsymbol{\alpha} \neq \boldsymbol{0}$) on the inputs between sampling instants. The matrix exponential $\boldsymbol{\Phi}_s = e^{\boldsymbol{A}\tau_s}$ can be computed in several different ways, but in general very effectively by means of a Padé approximation with repeated scaling and squaring (Moler & van Loan, 1978). However, both $\boldsymbol{\Phi}_s$ and the integral in (22) can be computed simultaneously through

$$\exp\left(\begin{bmatrix} -\boldsymbol{A} & \boldsymbol{\sigma}\boldsymbol{\sigma}^{\mathrm{T}} \\ \boldsymbol{0} & \boldsymbol{A}^{\mathrm{T}} \end{bmatrix} \tau_s = \begin{bmatrix} \boldsymbol{H}_1(\tau_s) & \boldsymbol{H}_2(\tau_s) \\ \boldsymbol{0} & \boldsymbol{H}_3(\tau_s) \end{bmatrix}\right) \tag{24}$$

by combining submatrices of the result (van Loan, 1978):

$$\boldsymbol{\Phi}_s = \boldsymbol{H}_3^{\mathrm{T}}(\tau_s), \tag{25}$$

$$\int_0^{\tau_s} e^{\boldsymbol{A}s}\boldsymbol{\sigma}\boldsymbol{\sigma}^{\mathrm{T}} e^{\boldsymbol{A}s^{\mathrm{T}}}\, ds = \boldsymbol{H}_3^{\mathrm{T}}(\tau_s)\boldsymbol{H}_2(\tau_s). \tag{26}$$

Remark 3. Solution (21) to (19) is undefined if $\boldsymbol{A}$ is singular, but by introducing a coordinate transformation based on the SVD of $\boldsymbol{A}$ a solution to (19) can also be found for singular $\boldsymbol{A}$ (Kristensen & Madsen, 2003).

2.2.2. *Maximum a posteriori estimation*

If prior information about the parameters is available in terms of a prior probability density function $p(\boldsymbol{\theta})$ for the parameters, Bayes' rule can be applied to give an improved estimate of the parameters by forming the posterior probability density function

$$p(\boldsymbol{\theta}|\mathscr{Y}_N) = \frac{p(\mathscr{Y}_N|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathscr{Y}_N)} \propto p(\mathscr{Y}_N|\boldsymbol{\theta})p(\boldsymbol{\theta}) \tag{27}$$

and subsequently finding the parameters that maximize this function, i.e. by performing *maximum a posteriori* (MAP) estimation. Assuming that the prior probability density of the parameters is Gaussian, and introducing

$$\boldsymbol{\mu}_{\boldsymbol{\theta}} = E\{\boldsymbol{\theta}\}, \tag{28}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{\theta}} = V\{\boldsymbol{\theta}\} \tag{29}$$

and

$$\boldsymbol{\varepsilon}_{\boldsymbol{\theta}} = \boldsymbol{\theta} - \boldsymbol{\mu}_{\boldsymbol{\theta}}, \tag{30}$$

the posterior probability density function becomes

$$p(\boldsymbol{\theta}|\mathscr{Y}_N) \propto \left(\prod_{k=1}^{N} \frac{\exp\left(-\frac{1}{2}\boldsymbol{\varepsilon}_k^{\mathrm{T}}\boldsymbol{R}_{k|k-1}^{-1}\boldsymbol{\varepsilon}_k\right)}{\sqrt{\det(\boldsymbol{R}_{k|k-1})}\left(\sqrt{2\pi}\right)^l}\right) p(\boldsymbol{y}_0|\boldsymbol{\theta}) \frac{\exp\left(-\frac{1}{2}\boldsymbol{\varepsilon}_{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}\boldsymbol{\varepsilon}_{\boldsymbol{\theta}}\right)}{\sqrt{\det(\boldsymbol{\Sigma}_{\boldsymbol{\theta}})}\left(\sqrt{2\pi}\right)^p} \tag{31}$$

and the parameter estimates can be determined by conditioning on $\boldsymbol{y}_0$ and solving the optimization problem

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta}\{-\ln(p(\boldsymbol{\theta}|\mathscr{Y}_N, \boldsymbol{y}_0))\}. \tag{32}$$

Remark 4. If no prior information is available (with $p(\boldsymbol{\theta})$ uniform), this formulation reduces to the ML formulation in (10). Thus MAP estimation can be seen as a generalization of ML estimation, which increases the flexibility of the estimation scheme. In fact, the formulation also allows MAP estimation on only some of the parameters (with $p(\boldsymbol{\theta})$ partly uniform), which increases the flexibility of the estimation scheme even further.

2.2.3. *Using multiple independent data sets*

If, instead of a single sequence of measurements, multiple consecutive, but separate, sequences of measurements, i.e. $\mathscr{Y}_{N_1}^1, \mathscr{Y}_{N_2}^2, \ldots, \mathscr{Y}_{N_i}^i, \ldots, \mathscr{Y}_{N_S}^S$, possibly of varying length, are available, a similar estimation method can be applied by expanding the expression for the posterior probability density function to the general form

$$p(\boldsymbol{\theta}|\mathbf{Y}) \propto \prod_{i=1}^{S}\left(\prod_{k=1}^{N_i} \frac{\exp\left(-\frac{1}{2}(\boldsymbol{\varepsilon}_k^i)^{\mathrm{T}}(\boldsymbol{R}_{k|k-1}^i)^{-1}\boldsymbol{\varepsilon}_k^i\right)}{\sqrt{\det(\boldsymbol{R}_{k|k-1}^i)}\left(\sqrt{2\pi}\right)^l}\right) p(\boldsymbol{y}_0^i|\boldsymbol{\theta}) \frac{\exp\left(-\frac{1}{2}\boldsymbol{\varepsilon}_{\boldsymbol{\theta}}^{\mathrm{T}}\boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}\boldsymbol{\varepsilon}_{\boldsymbol{\theta}}\right)}{\sqrt{\det(\boldsymbol{\Sigma}_{\boldsymbol{\theta}})}\left(\sqrt{2\pi}\right)^p}, \tag{33}$$

where

$$\mathbf{Y} = [\mathscr{Y}_{N_1}^1, \mathscr{Y}_{N_2}^2, \ldots, \mathscr{Y}_{N_i}^i, \ldots, \mathscr{Y}_{N_S}^S] \tag{34}$$

and assuming the individual sequences of measurements to be stochastically independent. The parameter estimates can now be determined by conditioning on

$$\mathbf{y_0} = [\boldsymbol{y}_0^1, \boldsymbol{y}_0^2, \ldots, \boldsymbol{y}_0^i, \ldots, \boldsymbol{y}_0^S] \tag{35}$$

and applying nonlinear optimization to find the minimum of the negative logarithm of the resulting posterior probability density function, i.e.

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\Theta}\{-\ln(p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{y_0}))\}. \tag{36}$$

Remark 5. If only one sequence of measurements is available ($S = 1$), this formulation reduces to the MAP formulation in (32), and it can therefore be seen as a generalization of the MAP formulation, which further increases the flexibility of the estimation scheme.

2.3. *Optimization issues*

To solve the nonlinear optimization problem (36) a quasi-Newton method based on the BFGS updating formula and a soft line search algorithm is applied within the software implementation of the proposed estimation scheme (see Section 3). This method is similar to the one presented by Dennis and Schnabel (1983), except for the fact that the gradient of the objective function here is approximated by a set of finite difference derivatives. During the initial iterations of the optimization algorithm, *forward differences* are used, but as the minimum of the objective function is approached the algorithm shifts to *central differences* to reduce the error of the approximation. In order to ensure stability in the calculation of the objective function in (36), simple constraints on the parameters are introduced, i.e.

$$\theta_j^{\min} < \theta_j < \theta_j^{\max}, \quad j = 1, \ldots, p. \tag{37}$$

These constraints are satisfied by solving with respect to a transformation of the original parameters, i.e.

$$\tilde{\theta}_j = \ln\left(\frac{\theta_j - \theta_j^{\min}}{\theta_j^{\max} - \theta_j}\right), \quad j = 1, \ldots, p. \tag{38}$$

This transformation does not influence the results of the estimation when θ_j is well within the imposed limits, because the estimator in (36) is invariant to conformal mappings such as the one in (38). However, a problem arises when θ_j is close to one of the limits, because the finite difference derivative with respect to θ_j may be close to zero. This problem is solved by adding a penalty function to (36) to give the modified objective function

$$\mathscr{F}(\boldsymbol{\theta}) = -\ln(p(\boldsymbol{\theta}|\mathbf{Y}, \mathbf{y_0})) + P(\lambda, \boldsymbol{\theta}, \boldsymbol{\theta}^{\min}, \boldsymbol{\theta}^{\max}) \tag{39}$$

which is used instead. The penalty function is given by

$$P(\lambda, \boldsymbol{\theta}, \boldsymbol{\theta}^{\min}, \boldsymbol{\theta}^{\max}) = \lambda\left(\sum_{j=1}^{p} \frac{|\theta_j^{\min}|}{\theta_j - \theta_j^{\min}} + \sum_{j=1}^{p} \frac{|\theta_j^{\max}|}{\theta_j^{\max} - \theta_j}\right) \tag{40}$$

for $|\theta_j^{\min}| > 0$ and $|\theta_j^{\max}| > 0, j=1, \ldots, p$. For proper choices of the Lagrange multiplier λ and the limiting values $\theta_j^{\min}$ and $\theta_j^{\max}$ the penalty function has no influence on the estimation for θ_j well within the limits, but forces the derivative to increase for θ_j close to the limits.

2.4. Data issues

Raw data sequences are often difficult to use for identification and parameter estimation, e.g. if irregular sampling has been applied, if there are occasional outliers or if some of the observations are missing.

The software implementation of the proposed estimation scheme (see Section 3) also provides features to deal with these issues, making it very flexible with respect to the types of data that can be used for the estimation.

2.4.1. Irregular sampling

The fact that the system equation (1) is continuous makes it easy to deal with irregular sampling, because the state prediction equations (17) and (18) of the EKF can be solved over time intervals of varying length.

2.4.2. Occasional outliers

The objective function (33) of the general formulation in (36) is quadratic in the innovations $\boldsymbol{\varepsilon}_k^i$, and this means that the corresponding parameter estimates are heavily influenced by occasional outliers in the data sets used for the estimation. To deal with this problem a robust estimation method is applied, where the objective function is modified by replacing the quadratic term

$$v_k^i = (\boldsymbol{\varepsilon}_k^i)^{\mathrm{T}} (\boldsymbol{R}_{k|k-1}^i)^{-1} \boldsymbol{\varepsilon}_k^i \tag{41}$$

with a function $\varphi(v_k^i)$, which returns the argument for small v_k^i, but is a linear function of $\boldsymbol{\varepsilon}_k^i$ for large v_k^i, i.e.

$$\varphi(v_k^i) = \begin{cases} v_k^i, & v_k^i < c^2, \\ c\left(2\sqrt{v_k^i} - c\right), & v_k^i \geqslant c^2, \end{cases} \tag{42}$$

where $c > 0$ is a constant. The derivative of this function with respect to $\boldsymbol{\varepsilon}_k^i$ is *Huber's ψ-function* (Huber, 1981).

2.4.3. Missing observations

The algorithms of the proposed estimation scheme make it easy to handle missing observations, i.e. missing values in the output vector $\boldsymbol{y}_k^i$, when calculating the term

$$\kappa_k^i = \frac{\exp\left(-\frac{1}{2}(\boldsymbol{\varepsilon}_k^i)^{\mathrm{T}} (\boldsymbol{R}_{k|k-1}^i)^{-1} \boldsymbol{\varepsilon}_k^i\right)}{\sqrt{\det(\boldsymbol{R}_{k|k-1}^i)}\left(\sqrt{2\pi}\right)^l} \tag{43}$$

in (33) for some i and some k. The usual way to account for missing or noninformative values in the EKF is to set the corresponding elements of the covariance matrix $\boldsymbol{S}$ in (12) to infinity, which in turn gives zeroes in the corresponding elements of $(\boldsymbol{R}_{k|k-1})^{-1}$ and the Kalman gain matrix $\boldsymbol{K}_k$, meaning that no updating will take place in (15) and (16) corresponding to the missing values.

This approach cannot be used for calculating (43), because a solution is needed which modifies $\boldsymbol{\varepsilon}_k^i$ and $\boldsymbol{R}_{k|k-1}^i$ to reflect that the effective dimension of $\boldsymbol{y}_k^i$ is reduced due to the missing values. This is accomplished by replacing (2) with the alternative measurement equation

$$\bar{\boldsymbol{y}}_k = \boldsymbol{E}(\boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k), \tag{44}$$

where $\boldsymbol{E}$ is an appropriate permutation matrix, which can be constructed from a unit matrix by eliminating the rows corresponding to missing values in $\boldsymbol{y}_k$. If, for example, $\boldsymbol{y}_k$ has three elements, and the middle one is missing, the appropriate permutation matrix is

$$\boldsymbol{E} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{45}$$

Equivalently, the equations of the EKF are replaced with the alternative output prediction equations

$$\hat{\bar{\boldsymbol{y}}}_{k|k-1} = \boldsymbol{E}\boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1}, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}), \tag{46}$$

$$\bar{\boldsymbol{R}}_{k|k-1} = \boldsymbol{E}\boldsymbol{C}\boldsymbol{P}_{k|k-1}\boldsymbol{C}^{\mathrm{T}}\boldsymbol{E}^{\mathrm{T}} + \boldsymbol{E}\boldsymbol{S}\boldsymbol{E}^{\mathrm{T}}, \tag{47}$$

the alternative innovation equation

$$\bar{\boldsymbol{\varepsilon}}_k = \overline{\boldsymbol{y}_k} - \hat{\bar{\boldsymbol{y}}}_{k|k-1}, \tag{48}$$

the alternative Kalman gain equation

$$\bar{\boldsymbol{K}}_k = \boldsymbol{P}_{k|k-1}\boldsymbol{C}^{\mathrm{T}}\boldsymbol{E}^{\mathrm{T}}\bar{\boldsymbol{R}}_{k|k-1}^{-1} \tag{49}$$

and the alternative updating equations

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \bar{\boldsymbol{K}}_k\bar{\boldsymbol{\varepsilon}}_k, \tag{50}$$

$$\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1} - \bar{\boldsymbol{K}}_k\bar{\boldsymbol{R}}_{k|k-1}\bar{\boldsymbol{K}}_k^{\mathrm{T}}. \tag{51}$$

The state prediction equations remain the same, and this in turn provides the necessary modifications of (43) to

$$\kappa_k^i = \frac{\exp\left(-\frac{1}{2}(\bar{\boldsymbol{\varepsilon}}_k^i)^{\mathrm{T}}(\bar{\boldsymbol{R}}_{k|k-1}^i)^{-1}\bar{\boldsymbol{\varepsilon}}_k^i\right)}{\sqrt{\det(\bar{\boldsymbol{R}}_{k|k-1}^i)}\left(\sqrt{2\pi}\right)^{\bar{l}}}, \tag{52}$$

where $\bar{l}$ is l minus the number of missing values in $\boldsymbol{y}_k^i$.

2.5. Uncertainty of parameter estimates

Essential outputs of any statistical parameter estimation scheme include an assessment of the uncertainty of the estimates and quantities facilitating subsequent statistical tests. Within the software implementation of the proposed estimation scheme (see Section 3), an estimate of the uncertainty of the parameter estimates is obtained by using the fact that by the central limit theorem the estimator in (36) is asymptotically Gaussian with mean $\boldsymbol{\theta}$ and covariance matrix

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}} = \boldsymbol{H}^{-1}, \tag{53}$$

where the matrix $\boldsymbol{H}$ is given by

$$h_{ij} = -E\left\{\frac{\partial^2}{\partial\theta_i\partial\theta_j}\ln(p(\boldsymbol{\theta}|\mathbf{Y},\mathbf{y_0}))\right\}, \quad i,j = 1,\ldots,p$$

and where an approximation to $\boldsymbol{H}$ can be obtained from

$$h_{ij} \approx -\left(\frac{\partial^2}{\partial\theta_i\partial\theta_j}\ln(\boldsymbol{p}(\boldsymbol{\theta}|\mathbf{Y},\mathbf{y_0}))\right)\Bigg|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}, \quad i,j = 1,\ldots,p.$$

The asymptotic Gaussianity of the estimator in (36) also allows t-tests to be performed to test whether a given parameter is marginally insignificant or not. The test quantity is the value of the parameter estimate divided by the standard deviation of the estimate, and under H_0 this quantity is asymptotically t-distributed with a number of degrees of freedom that equals the total number of observations minus the number of estimated parameters.

3. Software implementation

The parameter estimation scheme presented in Section 2 has been implemented in a software tool called CTSM, which is available for both Linux, Solaris and Windows.

3.1. Features

Within the graphical user interface (GUI) of CTSM, unknown parameters of model structures of the type in (1) and (2) can be estimated using the methods presented in Section 2. Once a model structure has been set up within the GUI, the program analyzes the model equations to determine the symbolic names of the parameters and displays them to allow the user to specify which parameters to fix, which to estimate, and how each parameter should be estimated (ML or MAP). The program automatically generates and compiles the FORTRAN-code needed to perform the estimation, including the code for obtaining the Jacobians needed for linearization of the nonlinear equations (through analytical manipulation of the FORTRAN-code in a pre-compiler to avoid numerical approximation). After specifying which data sets to use, the program determines the parameter estimates and displays them along with the statistics mentioned in Section 2. The program is very flexible with respect to the data sets that can be used for the estimation, because the features presented in Section 2 for dealing with irregular sampling, occasional outliers and missing observations have all been implemented as well.

3.2. Shared memory parallelization

Estimating parameters in grey-box models is a computationally demanding task in general, and the estimation scheme presented in Section 2 is no exception. On Solaris systems CTSM therefore supports shared memory parallelization using the OpenMP application program interface (API) by allowing the finite difference derivatives of the objective function constituting the gradient approximation to be computed in parallel.

Fig. 1 shows the performance benefits of this approach in terms of reduced execution time and demonstrates the scalability of the program for a small problem with 11 unknown parameters. The nonexisting effect of adding CPU's in the interval 6–10 is due to an uneven distribution of the workload (at least one CPU performs two finite difference computations, while the others wait), while for 11 and more CPUs the distribution is optimal.

4. Comparison with another software tool

A parameter estimation scheme rather similar to the one presented here and an associated software tool has previously been presented by Bohlin and Graebe (1995).

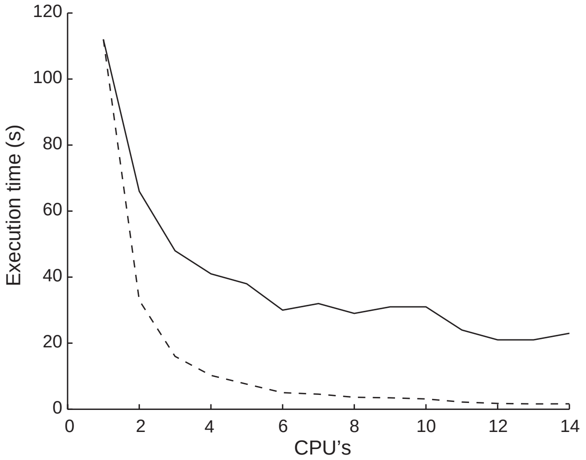


Fig. 1. Performance and scalability of CTSM when using shared memory parallelization. Solid lines: CTSM values; dashed lines: theoretical values (linear scalability).

There are, however, a number of very important differences between the two schemes, and this section is therefore devoted to outlining these differences and illustrating their influence on the estimation performance of the corresponding software tools through simulation studies. As mentioned in Section 3 the estimation scheme presented here has been implemented in a tool called CTSM. The original tool incorporating the scheme of Bohlin and Graebe (1995) was called IdKit, but has been further developed into a more extensive tool called MoCaVa (Bohlin, 2001), which runs under MATLAB.

Apart from parameter estimation, MoCaVa facilitates numerous other important tasks within grey-box model development, e.g. model validation, and is superior to CTSM in that respect. The latter only allows state and output predictions to be computed based on a given data set, whereas the former has various test and visualization features that allow a given model to be tested on another data set or against other models using the same data set. In fact, the essence of MoCaVa is the ability to iteratively develop unfalsified models by means of such techniques, or, more specifically, by means of a method based on the stepwise forward inclusion rule and a modified likelihood ratio statistic (Bohlin & Graebe, 1995; Bohlin, 2001). For the purpose of the following comparison with CTSM, however, only parameter estimation will be considered, since this constitutes a fundamental information generating task, upon which subsequent model development can be based.

4.1. Mathematical and algorithmic differences

Although very similar in terms of parameter estimation algorithms, there are some distinct differences between MoCaVa and CTSM. Generally, MoCaVa has more restrictions and uses more crude approximations than CTSM, which reduces the computational load, but at the expense of accuracy and consistency.

4.1.1. General model structure

With respect to the general model structure MoCaVa is less flexible than CTSM, primarily with respect to the diffusion and measurement noise terms. Within IdKit the following class of models was allowed:

$$\mathrm{d}\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})\,\mathrm{d}t + \boldsymbol{\sigma}(t, \boldsymbol{\theta})\,\mathrm{d}\boldsymbol{\omega}_t, \tag{54}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k, \tag{55}$$

where $\boldsymbol{e}_k \in \mathcal{N}(\boldsymbol{0}, \boldsymbol{S}(t_k, \boldsymbol{\theta}))$, i.e. almost the same class of models as in CTSM, but within MoCaVa this class has been restricted to the following:

$$\mathrm{d}\boldsymbol{x}_t = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t, t, \boldsymbol{\theta})\,\mathrm{d}t, \tag{56}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k, t_k, \boldsymbol{\theta}) + \boldsymbol{e}_k, \tag{57}$$

where $\boldsymbol{e}_k \in \mathcal{N}(\boldsymbol{0}, \boldsymbol{S}(\boldsymbol{\theta}))$ and $\boldsymbol{S}$ is a diagonal matrix. In other words, no diffusion term is allowed and there are more restrictions on the parametrization of the measurement noise term. This limits flexibility, but instead some of the inputs can be modelled as disturbances and a library of generic disturbance models is provided. E.g. Bohlin (2001) argues that moderate diffusion may be well approximated by a low-pass filtered white noise disturbance with a bandwidth below the Nyquist frequency.

4.1.2. Parameter estimation methods

With respect to parameter estimation methods, both programs provide a ML/MAP estimation setup, but MoCaVa does not allow estimation on multiple independent data sets as is the case with CTSM. Furthermore, the specific implementations of the ML estimation setup differ, although both programs rely on the same assumption of Gaussianity of the innovations and use the EKF to compute them. This is due to some important differences in the implementations of the EKF. MoCaVa uses an approach very similar to the linearization-based approach in CTSM, but with a more crude first order Taylor approximation to the matrix exponential. In addition, since diffusion terms are not allowed in the general model structure in MoCaVa, it suffices to compute the exponential of a much simpler matrix than in CTSM. More importantly, however, like the original IdKit program, MoCaVa obtains the Jacobians needed for linearization of the nonlinear equations by making finite difference approximations around a reference trajectory obtained by applying the EKF without updating. Thus the original equations are not linearized at points corresponding to the current state estimates, but at points along a deterministic reference trajectory. This is a very important difference from CTSM, which renders IdKit and hence MoCaVa unsuitable for estimation of parameters in nonlinear systems with significant diffusion (Bohlin & Graebe, 1995; Bohlin, 2001). This is demonstrated below.

4.1.3. Optimization issues

There are also some important differences between the two programs with respect to optimization method. CTSM uses a quasi-Newton method based on the BFGS updating formula for the Hessian and a soft line search algorithm, whereas MoCaVa uses a modified Newton-Raphson method, where the Hessian is approximated by applying a statistical assumption (Bohlin, 2001). Both programs use finite differences to approximate the gradient of the objective function, but while CTSM shifts from forward to central differences as the minimum of the objective function is approached MoCaVa only uses forward differences. Finally, and perhaps most importantly, the termination criterion in CTSM is a function of the relative reduction in the objective function as well as the relative change in the parameter values, whereas in MoCaVa it is only a function of the relative reduction in the objective function.

4.1.4. Data issues

In terms of flexibility with respect to the types of data that can be used for the estimation, the two programs are almost equivalent. The only important difference is that MoCaVa does not incorporate any outlier robustness features, but relies on the user to remove outliers.

4.1.5. Uncertainty of parameter estimates

As opposed to CTSM, where an assessment of the uncertainty of the parameter estimates is obtained, no such information is obtained directly in MoCaVa.

4.2. Simulation studies

In the following some of the effects of the differences between MoCaVa and CTSM are illustrated by means of estimation results from two simulation examples.

Example 1: *Nonlinear* (*NL*) *model*: The first example used is a simple model of a fed-batch bioreactor. The system equation of the model is

$$\mathrm{d}\begin{pmatrix} X \\ S \\ V \end{pmatrix} = \begin{pmatrix} \mu(S)X - \frac{FX}{V} \\ \frac{-\mu(S)X}{Y} + \frac{F(S_F - S)}{V} \\ F \end{pmatrix} \mathrm{d}t + \begin{bmatrix} \sigma_{11} & 0 & 0 \\ 0 & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix} \mathrm{d}\boldsymbol{\omega}_t,$$

where X is the biomass concentration, S is the substrate concentration, V is the volume, F is the feed flow rate, $Y = 0.5$ is a yield coefficient, $S_F = 10$ is the feed concentration, and the growth rate $\mu(S)$ is given by

$$\mu(S) = \mu_{\max} \frac{S}{K_2 S^2 + S + K_1}$$

with $\mu_{\max}$, K_1 and $K_2 = 0.5$ as kinetic parameters. The measurement equation of the model is

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix}_k = \begin{pmatrix} X \\ S \\ V \end{pmatrix}_k + \boldsymbol{e}_k,$$

where $\boldsymbol{e}_k \in \mathcal{N}(\boldsymbol{0}, \boldsymbol{S})$ and where

$$\boldsymbol{S} = \begin{bmatrix} S_{11} & 0 & 0 \\ 0 & S_{22} & 0 \\ 0 & 0 & S_{33} \end{bmatrix}.$$

Using the true parameters and initial states shown in Tables 1–3 three different sets of data (each 101 samples over 3.8 h) were generated by stochastic simulation using the Euler scheme (Kloeden & Platen, 1992):

(1) A data set with no diffusion (Fig. 2a).
(2) A data set with weak diffusion (Fig. 2b).
(3) A data set with strong diffusion (Fig. 2c).

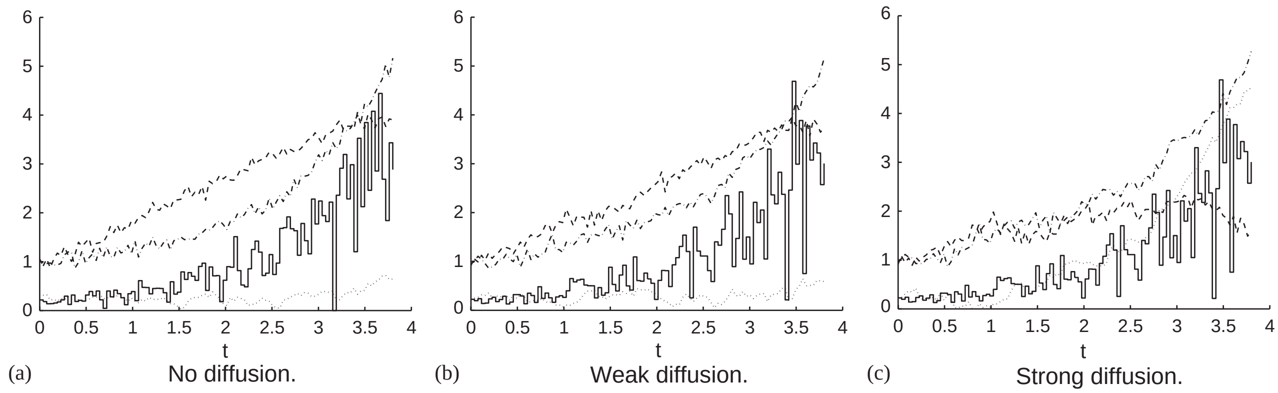


Fig. 2. Simulated data sets for Example 1. Solid staircase: F; dashed lines: y_1; dotted lines: y_2; dash-dotted lines: y_3. (a) No diffusion; (b) weak diffusion; (c) strong diffusion.

Table 1
Estimation results. Example 1—Data in Fig. 2a

Parameter	True value	CTSM	MoCaVa
X_0	1.0000E+00	1.0081E+00	9.9187E-01
S_0	2.4495E-01	2.5160E-01	2.3371E-01
V_0	1.0000E+00	1.0007E+00	9.9533E-01
$\mu_{\max}$	1.0000E+00	1.0104E+00	1.0143E+00
K_1	3.0000E-02	3.4177E-02	3.7176E-02
σ_{11}	0.0000E+00	6.8942E-06	9.9095E-03
σ_{22}	0.0000E+00	4.2411E-07	9.9727E-03
σ_{33}	0.0000E+00	5.1325E-07	9.7394E-03
S_{11}	1.0000E-02	9.0855E-03	8.6565E-03
S_{22}	1.0000E-03	9.7370E-04	9.4740E-04
S_{33}	1.0000E-02	9.4517E-03	8.9991E-03

Table 2
Estimation results. Example 1—Data in Fig. 2b

Parameter	True value	CTSM	MoCaVa
X_0	1.0000E+00	9.8615E-01	9.9193E-01
S_0	2.4495E-01	2.3800E-01	2.3159E-01
V_0	1.0000E+00	9.7733E-01	1.0694E+00
$\mu_{\max}$	1.0000E+00	9.9694E-01	9.5656E-01
K_1	3.0000E-02	3.1506E-02	2.7128E-02
σ_{11}	1.0000E-01	1.1782E-01	3.0813E-01
σ_{22}	1.0000E-01	7.8251E-02	1.0167E-02
σ_{33}	1.0000E-01	6.2429E-02	1.0025E-02
S_{11}	1.0000E-02	8.0729E-03	9.2114E-03
S_{22}	1.0000E-03	9.2753E-04	1.2410E-03
S_{33}	1.0000E-02	9.3570E-03	1.2237E-02

Table 3
Estimation results. Example 1—Data in Fig. 2c

Parameter	True value	CTSM	MoCaVa
X_0	1.0000E+00	9.6106E-01	9.5386E-01
S_0	2.4495E-01	2.3457E-01	1.0003E-01
V_0	1.0000E+00	9.9349E-01	1.0368E+00
$\mu_{\max}$	1.0000E+00	9.7142E-01	9.0460E-01
K_1	3.0000E-02	3.2600E-02	1.9886E-02
σ_{11}	3.1623E-01	3.2500E-01	1.1169E+00
σ_{22}	3.1623E-01	2.8063E-01	1.0046E-02
σ_{33}	3.1623E-01	2.6078E-01	5.5165E-01
S_{11}	1.0000E-02	7.7174E-03	9.9452E-03
S_{22}	1.0000E-03	1.1618E-03	1.1330E-02
S_{33}	1.0000E-02	8.3037E-03	1.5597E-02

Example 2: *Linear time-invariant* (*LTI*) *model*: The second example used is a simple second order lumped parameter model of the heat dynamics of a wall with system equation

$$\mathrm{d}\begin{pmatrix} T_1 \\ T_2 \end{pmatrix} = \left(\begin{bmatrix} -\frac{1}{G_1}\left(\frac{1}{H_1}+\frac{1}{H_2}\right) & \frac{1}{G_1H_2} \\ \frac{1}{G_2H_2} & -\frac{1}{G_2}\left(\frac{1}{H_2}+\frac{1}{H_3}\right) \end{bmatrix} \begin{pmatrix} T_1 \\ T_2 \end{pmatrix} + \begin{bmatrix} \dfrac{1}{G_1H_1} & 0 \\ 0 & \dfrac{1}{G_2H_3} \end{bmatrix} \begin{pmatrix} T_\mathrm{e} \\ T_\mathrm{i} \end{pmatrix} \right) \mathrm{d}t + \begin{bmatrix} \sigma_{11} & 0 \\ 0 & \sigma_{22} \end{bmatrix} \mathrm{d}\boldsymbol{\omega}_t,$$

where T_1 is the outer wall temperature, T_2 is the inner wall temperature, T_e is the outdoor temperature, T_i is the indoor temperature, and G_1, G_2, H_1, H_2 and H_3 are parameters of the second order thermal network describing the wall. The measurement equation is

$$(q_i)_k = \begin{bmatrix} 0 & -\frac{1}{H_3} \end{bmatrix} \begin{pmatrix} T_1 \\ T_2 \end{pmatrix}_k + \begin{bmatrix} 0 & \frac{1}{H_3} \end{bmatrix} \begin{pmatrix} T_\mathrm{e} \\ T_\mathrm{i} \end{pmatrix}_k + e_k,$$

where $e_k \in \mathcal{N}(0, S)$. Using the true parameters and initial states shown in Tables 4 and 5 two different sets of data (each 719 samples over 718 h) were generated by stochastic simulation using the method from Example 1:

(1) A data set without diffusion (Fig. 3a).
(2) A data set with diffusion (Fig. 3b).

4.2.1. *Quality of estimates*

The first issue addressed in the comparison of the estimation performance of MoCaVa and CTSM is quality of estimates. This should ideally be based on extensive Monte Carlo simulation analysis, which would deliver an assessment of both bias and variance of both estimators. However, both CTSM and MoCaVa are interactive programs in the sense that user intervention is required to process each data set, and such analysis is therefore prohibitively time-consuming given the number of data sets to be processed to obtain a reliable assessment. Instead, the two programs are compared in terms of single estimation error using the data sets mentioned above.

Tables 1–3 show estimation results from both programs for the NL case in Example 1 using the data sets shown in Fig. 2 (zero order hold on input). For the estimation in MoCaVa the diffusion term was approximated by a low-pass filtered white noise disturbance with a bandwidth of 10 rad/h (the Nyquist frequency is about 82.7 rad/h). The results clearly show that the estimates obtained with CTSM have less error, in particular the estimates of the parameters of the diffusion term, some of which are an order of magnitude off in MoCaVa. Furthermore, the inability of MoCaVa to correctly estimate these parameters seems to introduce additional error in the estimates of the other parameters.

Tables 4 and 5 show estimation results for the LTI case in Example 2 using the data sets shown in Fig. 3 (zero order hold on input). For the estimation in MoCaVa the diffusion term was approximated by a low-pass filtered white noise disturbance with a bandwidth of 0.4 rad/h (the Nyquist frequency is about 3.14 rad/h). In this case much more similar estimates are obtained.

Table 4
Estimation results. Example 2—Data in Fig. 3a

Parameter	True value	CTSM	MoCaVa
T_{10}	1.3200E+01	1.3134E+01	1.3271E+01
T_{20}	2.5300E+01	2.5330E+01	2.5571E+01
G_1	1.0000E+02	1.0394E+02	1.0189E+02
G_2	5.0000E+01	4.9320E+01	4.9266E+01
H_1	1.0000E+00	9.6509E-01	9.8904E-01
H_2	2.0000E+00	2.0215E+00	1.9965E+00
H_3	5.0000E-01	5.0929E-01	5.0929E-01
σ_{11}	0.0000E+00	4.2597E-08	8.3838E-03
σ_{22}	0.0000E+00	1.4278E-09	5.1542E-03
S	1.0000E-02	1.0330E-02	1.0019E-02

Table 5
Estimation results. Example 2—Data in Fig. 3b

Parameter	True value	CTSM	MoCaVa
T_{10}	1.3200E+01	1.9541E+01	1.4851E+01
T_{20}	2.5300E+01	2.5360E+01	2.5580E+01
G_1	1.0000E+02	1.0718E+02	7.6394E+01
G_2	5.0000E+01	5.3125E+01	5.4272E+01
H_1	1.0000E+00	1.9902E+00	1.4285E+00
H_2	2.0000E+00	9.0621E-01	1.9034E+00
H_3	5.0000E-01	5.0844E-01	5.1010E-01
σ_{11}	1.0000E-01	1.7791E-01	1.0206E-02
σ_{22}	1.0000E-01	1.4951E-01	1.4089E-01
S	1.0000E-02	9.4965E-03	3.2529E-02

4.2.2. *Reproducibility*

The second issue addressed in the comparison of the estimation performance of the two programs is reproducibility in terms of the sensitivity of the results to variations in initial values for the optimization.

Tables 6 and 7 show estimation results from CTSM and MoCaVa, respectively, for the NL case corresponding to Table 1 using four different sets of initial values. The initial values used are the true values shown in Table 1, except for the values of the parameters of the diffusion term, which have been varied ($[1, 0.1, 0.01, 0.001]$). The results clearly show that MoCaVa is much more sensitive than CTSM to variations in initial values, particularly with respect to the parameters of the diffusion term.

Tables 8 and 9 show equivalent results for the LTI case corresponding to Table 4. The initial values used in this case are the true values shown in Table 4, except for the values of the parameters of the diffusion term, which have been varied ($[1, 0.1, 0.01, 0.001]$). Note that for the first set of initial values, MoCaVa was not able to converge. Again the results show that MoCaVa is more sensitive than CTSM, particularly with respect to the parameters of the diffusion term.

5. Discussion

The results presented in Section 4 show that the software tool presented in Section 3 for estimation of parameters in grey-box models (CTSM) generally performs well. In particular, it performs better than the one presented by Bohlin (2001) (MoCaVa) due to a number of algorithmic differences between the two programs.

In terms of quality of estimates, CTSM clearly gives less error than MoCaVa for nonlinear systems with significant diffusion, especially with respect to the parameters of the diffusion term. It may be argued that this is due to the approximation used in MoCaVa, because the diffusion term cannot be modelled explicitly, and hence that a comparison should have been made with the original IdKit program by Bohlin and Graebe (1995), but this program is not readily available. Furthermore, Bohlin and Graebe (1995) argue that IdKit cannot be expected to work properly for models with significant diffusion, so the differences in results from CTSM may be due to the construction of the algorithms after all. With respect to the quality of the estimates of the parameters of the diffusion term, it is particularly important that the EKF implementation in CTSM uses analytical Jacobians obtained at current values of the state estimates, whereas MoCaVa uses numerical Jacobians obtained at state values along a deterministic reference trajectory. This

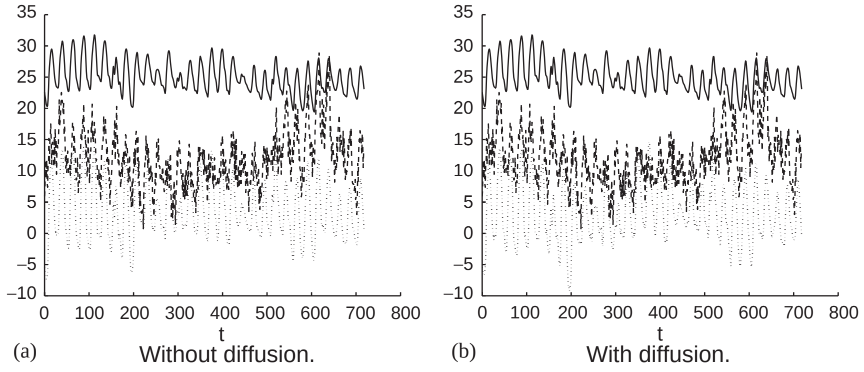


Fig. 3. Simulated data sets for Example 2. Solid lines: T_i; dashed lines: T_e; dotted lines: q_i. (a) Without diffusion; (b) with diffusion.

Table 6
CTSM reproducibility. Example 1—Data in Fig. 2a

Parameter	Result 1	Result 2	Result 3	Result 4
X_0	1.0081E+00	1.0081E+00	1.0081E+00	1.0086E+00
S_0	2.5160E-01	2.5160E-01	2.5160E-01	2.5205E-01
V_0	1.0007E+00	1.0007E+00	1.0007E+00	1.0006E+00
μ_{max}	1.0104E+00	1.0104E+00	1.0104E+00	1.0107E+00
K_1	3.4178E-02	3.4177E-02	3.4177E-02	3.4289E-02
σ_{11}	2.7167E-08	6.5411E-06	6.8942E-06	3.0674E-04
σ_{22}	3.5673E-06	8.7657E-18	4.2411E-07	5.9732E-05
σ_{33}	1.1250E-07	5.0250E-09	5.1325E-07	1.6944E-04
S_{11}	9.0855E-03	9.0855E-03	9.0855E-03	9.0844E-03
S_{22}	9.7371E-04	9.7370E-04	9.7370E-04	9.7068E-04
S_{33}	9.4517E-03	9.4517E-03	9.4517E-03	9.4239E-03

Table 7
MoCaVa reproducibility. Example 1—Data in Fig. 2a

Parameter	Result 1	Result 2	Result 3	Result 4
X_0	9.8736E-01	9.8528E-01	9.9187E-01	9.9247E-01
S_0	2.5036E-01	2.3963E-01	2.3371E-01	2.3351E-01
V_0	1.0027E+00	9.9632E-01	9.9533E-01	9.9527E-01
μ_{max}	1.0230E+00	1.0213E+00	1.0143E+00	1.0134E+00
K_1	3.7723E-02	3.7639E-02	3.7176E-02	3.7035E-02
σ_{11}	1.4692E-01	6.2238E-02	9.9095E-03	9.9963E-04
σ_{22}	1.5229E-01	7.7283E-02	9.9727E-03	1.0000E-03
σ_{33}	1.2476E-01	5.8497E-02	9.7394E-03	1.0022E-03
S_{11}	8.2961E-03	8.4638E-03	8.6565E-03	8.6720E-03
S_{22}	9.0169E-04	9.3558E-04	9.4740E-04	9.4002E-04
S_{33}	8.7933E-03	8.8285E-03	8.9991E-03	9.0133E-03

becomes particularly evident when comparing the results from the nonlinear model with the results from the linear time invariant model. In the nonlinear case CTSM performs better than MoCaVa, whereas the two programs perform equally well in the linear time invariant case, where the Jacobians are equal.

In terms of reproducibility, CTSM is less sensitive to initial values and hence gives more consistent results, which is most likely due to the fact that in MoCaVa the termination criterion for the optimization algorithm is only a function of the relative reduction in the objective function, whereas in CTSM it is also a function of the relative change in the parameter values. Evidence to support this conclusion is the fact that similar results have been obtained using data from a nonlinear and a linear time invariant system without diffusion, indicating that the result is independent of the system type and of the diffusion term approximation.

In the more general setting of providing support for systematic grey-box model development, MoCaVa is superior to CTSM, because of the many additional features included to facilitate various model development tasks. In this setting it may also be argued that the improvement in speed obtained through the more crude approximations made in MoCaVa is an advantage, but unfortunately this improvement seems to come at the price of accuracy for nonlinear systems with significant diffusion and of consistency, particularly with respect to the estimates of the parameters of the diffusion term. For applications where these estimates are used directly, e.g. to assess the quality of a model (Kristensen et al., 2001), to discriminate between different models (Kristensen et al., 2002), or to pinpoint model deficiencies (Kristensen et al., 2004), one cannot afford this.

6. Conclusion

An efficient and flexible scheme for parameter estimation in stochastic grey-box models has been presented. Based on the extended Kalman filter it features maximum likelihood as well as maximum a posteriori estimation on multiple independent data sets, including irregularly sampled data sets and data sets with occasional outliers and missing observations. A software tool implementing the estimation scheme has also been presented and a comparison with an existing tool has indicated that the new tool has better performance both in terms of quality of estimates for nonlinear systems with significant diffusion and in terms of reproducibility. In

Table 8
CTSM reproducibility. Example 2—Data in Fig. 3a

Parameter	Result 1	Result 2	Result 3	Result 4
T_{10}	1.3134E+01	1.3134E+01	1.3134E+01	1.3134E+01
T_{20}	2.5330E+01	2.5330E+01	2.5330E+01	2.5330E+01
G_1	1.0394E+02	1.0394E+02	1.0394E+02	1.0395E+02
G_2	4.9320E+01	4.9320E+01	4.9320E+01	4.9320E+01
H_1	9.6509E-01	9.6509E-01	9.6509E-01	9.6506E-01
H_2	2.0215E+00	2.0215E+00	2.0215E+00	2.0215E+00
H_3	5.0929E-01	5.0929E-01	5.0929E-01	5.0929E-01
σ_{11}	2.1538E-19	8.7694E-11	4.2597E-08	8.8565E-06
σ_{22}	3.4939E-08	5.5784E-08	1.4278E-09	3.0702E-07
S	1.0330E-02	1.0330E-02	1.0330E-02	1.0330E-02

Table 9
MoCaVa reproducibility. Example 2—Data in Fig. 3a

Parameter	Result 1	Result 2	Result 3	Result 4
T_{10}	—	1.3070E+01	1.3271E+01	1.3168E+01
T_{20}	—	2.5577E+01	2.5571E+01	2.5567E+01
G_1	—	1.0270E+02	1.0189E+02	1.0373E+02
G_2	—	4.9277E+01	4.9266E+01	4.9312E+01
H_1	—	9.5979E-01	9.8904E-01	9.6833E-01
H_2	—	2.0277E+00	1.9965E+00	2.0180E+00
H_3	—	5.0935E-01	5.0929E-01	5.0929E-01
σ_{11}	—	2.2435E-02	8.3838E-03	9.9907E-04
σ_{22}	—	7.9109E-03	5.1542E-03	1.0036E-03
S	—	9.9315E-03	1.0019E-02	1.0224E-02

particular, the new tool provides more accurate and more consistent estimates of the diffusion term parameters.

Acknowledgements

The authors would like to acknowledge Torsten Bohlin for his contribution to this paper, particularly for supplying and providing instructions for MoCaVa.

References

Allgöwer, F., & Zheng, A. (Eds.) (2000). *Nonlinear model predictive control*, *Progress in systems & control theory*, Vol. 26. Switzerland: Birkhauser Verlag.

Åström, K. J. (1970). *Introduction to stochastic control theory*. New York, USA: Academic Press.

Bak, J., Madsen, H., & Nielsen, H. A. (1999). Goodness of fit of stochastic differential equations. In P. Linde, & A. Holm (Eds.), *Symposium i Anvendt Statistik*. Copenhagen, Denmark: Copenhagen Business School.

Bitmead, R. R., Gevers, M., & Wertz, V. (1990). *Adaptive optimal control—the thinking man's GPC*. New York, USA: Prentice-Hall.

Bohlin, T. (2001). *A grey-box process identification tool: Theory and practice.* Technical Report IR-S3-REG-0103, Department of Signals, Sensors and Systems, Royal Institute of Technology, Stockholm, Sweden.

Bohlin, T., & Graebe, S. F. (1995). Issues in nonlinear stochastic grey-box identification. *International Journal of Adaptive Control and Signal Processing*, *9*, 465–490.

Dennis, J. E., & Schnabel, R. B. (1983). *Numerical methods for unconstrained optimization and nonlinear equations*. Englewood Cliffs, USA: Prentice-Hall.

Holst, J., Holst, U., Madsen, H., & Melgaard, H. (1992). Validation of grey box models. In L. Dugard, M. M'Saad, & I. D. Landau (Eds.), *Selected papers from the fourth IFAC symposium on adaptive systems in control and signal processing* (pp. 407–414). Oxford: Pergamon Press.

Huber, P. J. (1981). *Robust statistics*. New York, USA: Wiley.

Jacobsen, J. L., & Madsen, H. (1996). Grey box modelling of oxygen levels in a small stream. *Environmetrics*, *7*(1), 109–121.

Jazwinski, A. H. (1970). *Stochastic processes and filtering theory*. New York, USA: Academic Press.

Kloeden, P. E., & Platen, E. (1992). *Numerical solution of stochastic differential equations*. Berlin, Germany: Springer.

Kristensen, N. R., & Madsen, H. (2003). *Continuous time stochastic modelling—CTSM 2.2*. Technical University of Denmark, Lyngby, Denmark.

Kristensen, N. R., Madsen, H., & Jørgensen, S. B. (2001). Computer aided continuous time stochastic process modelling. In R. Gani, & S. B. Jørgensen (Eds.), *European symposium on computer aided process engineering*, Vol. 11. Amsterdam: Elsevier.

Kristensen, N. R., Madsen, H., & Jørgensen, S. B. (2002). Using continuous time stochastic modelling and nonparametric statistics to improve the quality of first principles models. In J. Grievink, & J. van Schijndel (Eds.), *European symposium on computer aided process engineering*, Vol. 12. Amsterdam: Elsevier.

Kristensen, N. R., Madsen, H., & Jørgensen, S. B. (2004). A method for systematic improvement of stochastic grey-box models. *Computers and Chemical Engineering* (in press).

Ljung, L. (1987). *System identification: Theory for the user*. New York, USA: Prentice-Hall.

Madsen, H., & Holst, J. (1995). Estimation of continuous-time models for the heat dynamics of a building. *Energy and Buildings*, *22*, 67–79.

Moler, C., & van Loan, C. F. (1978). Nineteen dubious ways to compute the exponential of a matrix. *SIAM Review*, *20*(4), 801–836.

Nielsen, J. N., & Madsen, H. (2001). Applying the EKF to stochastic differential equations with level effects. *Automatica*, *37*, 107–112.

Nielsen, J. N., Madsen, H., & Young, P. C. (2000). Parameter estimation in stochastic differential equations: An overview. *Annual Reviews in Control*, *24*, 83–94.

Söderström, T., & Stoica, P. (1989). *System identification*. New York, USA: Prentice-Hall.

Unbehauen, H., & Rao, G. P. (1990). Continuous-time approaches to system identification—A survey. *Automatica*, *26*(1), 23–35.

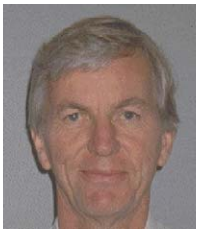
Unbehauen, H., & Rao, G. P. (1998). A review of identification in continuous-time systems. *Annual Reviews in Control*, *22*, 145–171.

van Loan, C. F. (1978). Computing integrals involving the matrix exponential. *IEEE Transactions on Automatic Control*, *23*(3), 395–404.

Young, P. C. (1981). Parameter estimation for continuous-time models—A survey. *Automatica*, *17*(1), 23–39.

Niels Rode Kristensen He received the M.Sc. in 1999, and the Ph.D. in 2003, both in Chemical Engineering at the Technical University of Denmark. His main research interest is grey-box modelling with stochastic differential equations and applications thereof, but other interests include time series analysis, stochastic filtering theory, estimation and control.

Henrik Madsen He received the M.Sc. in Engineering in 1982, and the Ph.D. in Statistics in 1986, both at the Technical University of Denmark. He was appointed Assistant Professor in Statistics in 1986, Associate Professor in 1989, and professor in Statistics with a special focus on Stochastic Dynamic Systems in 1999. He has been external lecturer at a number of universities. He is involved in a large number of cooperative projects with other universities, research organisations and industrial partners. His main research interest is related to analysis and modelling of stochastic dynamics systems. This includes signal processing, time series analysis, identification, estimation, grey-box modelling, prediction, optimization and control. The applications are mostly related to Energy Systems, Informatics, Environmental Systems, Bioinformatics, Process Modelling and Finance. He has authored or co-authored approximately 210 papers and technical reports, and about 10 educational texts. He is also the leader of *Center for High Performance Computing* at DTU which was opened by the Danish Minister of Research in February 2002.



Sten Bay Jørgensen He received the M.Sc. in 1963, and the Ph.D. in 1969, both in Chemical Engineering at the Technical University of Denmark. He was then a Research Associate at the Department of Chemical Engineering at Columbia University, New York, until he was appointed Associate Professor in Chemical Engineering at the Technical University of Denmark in 1971. In 1986 he was appointed Professor in Chemical Engineering with a special focus on Chemical Process Modelling, Dynamics, Identification and Control. His research interests also include Chemical Process and Product Design, and he is involved in a number of projects with other research organisations and industrial partners. The relevant industries include oil, petrochemical, chemical, bio-chemical and pharmaceutical. He has authored or co-authored a large number of papers and he is a reviewer for several journals within Chemical Engineering and Process Control.