PERGAMON

Automatica 36 (2000) 1199–1204

www.elsevier.com/locate/automatica

Brief Paper

Tracking time-varying parameters with local regression☆

Alfred Joensen*, Henrik Madsen, Henrik Aa. Nielsen, Torben S. Nielsen

Department of Mathematical Modelling, Technical University of Denmark, Bldg. 321, DK-2800 Lyngby, Denmark

Received 20 August 1998; revised 11 June 1999; received in final form 18 November 1999

Abstract

This paper shows that the recursive least-squares (RLS) algorithm with forgetting factor is a special case of a varying-coefficient model, and a model which can easily be estimated via simple local regression. This observation allows us to formulate a new method which retains the RLS algorithm, but extends the algorithm by including polynomial approximations. Simulation results are provided, which indicates that this new method is superior to the classical RLS method, if the parameter variations are smooth. © 2000 Elsevier Science Ltd. All rights reserved.

Keywords: Recursive estimation; Varying-coefficient; Conditional parametric; Polynomial approximation; Weighting functions

1. Introduction

The RLS algorithm with forgetting factor (Ljung & Soderstrom, 1983) is often applied in on-line situations, where time variations are not modeled adequately by a linear model. By sliding a time window of a specific width over the observations where only the newest observations are seen, the model is able to adapt to slow variations in the dynamics. The width, or the bandwidth h, of the time window determines how fast the model adapts to the variations, and the most adequate value of h depends on how fast the parameters actually vary in time. If the time variations are fast, h should be small, otherwise the estimates will be seriously biased. However, fast adaption means that only few observations are used for the estimation, which results in a noisy estimate. Therefore, the choice of h can be seen as a bias/variance trade off.

In the context of local regression (Cleveland & Devlin, 1988) the parameters of a linear model estimated by the RLS algorithm can be interpreted as zero-order local time polynomials, or in other words local constants. However, it is well known that polynomials of higher order in many cases provide better approximations than local constants. The objective of this paper is thus to illustrate the similarity between the RLS algorithm and local regression, which leads to a natural extension of the RLS algorithm, where the parameters are approximated by higher-order local time polynomials. This approach does, to some degree, represent a solution to the bias/variance trade off. Furthermore, viewing the RLS algorithm as local regression, could potentially lead to development of new and refined RLS algorithms, as local regression is an area of current and extensive research. A generalisation of models with varying parameters is presented in Hastie and Tibshirani (1993), and, as will be shown in this paper, the RLS algorithm is an estimation method for one of these models.

Several extensions of the RLS algorithm have been proposed in the literature, especially to handle situations where the parameter variations are not the same for all the parameters. Such situations can be handled by assigning individual bandwidths to each parameter, e.g. *vector forgetting*, or by using the *Kalman Filter* (Parkum, Poulsen & Holst, 1992). These approaches all have drawbacks, such as assumptions that the parameters are uncorrelated and/or are described by a random walk. Polynomial approximations and local regression can to some degree take care of these situations, by approximating the parameters with polynomials of different degrees. Furthermore, it is obvious that the parameters can be functions of other variables than time. In Nielsen, Nielsen, Madsen and Joensen (2000) a recursive algorithm is

☆This paper was not presented at any IFAC meeting. This paper was recommended for publication in revised form by Associate Editor Brett Ninness under the direction of Editor T. Söderström.

* Corresponding author. Tel.: + 45-4525-3411; fax: + 45-4588-1397. *E-mail address:* akj@imm.dtu.dk (A. Joensen).

0005-1098/00/$ - see front matter © 2000 Elsevier Science Ltd. All rights reserved.
PII: S0005-1098(00)00029-7

proposed, which can be used when the parameters are functions of time and some other explanatory variables.

Local regression is adequate when the parameters are functions of the same explanatory variables. If the parameters depend on individual explanatory variables, estimation methods for additive models should be used (Fan, Hardle & Mammer, 1998; Hastie & Tibshirani, 1990). Unfortunately it is not obvious how to formulate recursive versions of these estimation methods, and to the authors best knowledge no such recursive methods exists. Early work on additive models and recursive regression dates back to Holt (1957) and Winters (1960), which developed recursive estimation methods for models related to the additive models, where individual forgetting factors are assigned to each additive component, and the trend is approximated by a polynomial in time.

2. The varying-coefficient approach

Varying-coefficient models are considered in Hastie and Tibshirani (1993). These models can be considered as linear regression models in which the parameters are replaced by smooth functions of some explanatory variables. This section gives a short introduction to the varying-coefficient approach and a method of estimation, local regression, which becomes the background for the proposed extension of the RLS algorithm.

2.1. The model

We define the varying-coefficient model

$$y_i = \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{\theta}(\boldsymbol{x}_i) + e_i, \quad i = 1, \dots, N, \tag{1}$$

where y_i is a response, $\boldsymbol{x}_i$ and $\boldsymbol{z}_i$ are explanatory variables, $\boldsymbol{\theta}(\cdot)$ is a vector of unknown but smooth functions with values in $\boldsymbol{R}$, and N is the number of observations. If ordinary regression is considered e_i should be identically distributed (i.d.), but if i denotes at time index and $\boldsymbol{z}_i^{\mathrm{T}}$ contains lagged values of the response variable, e_i should be independent and identically distributed (i.i.d.).

The definition of a varying-coefficient model in Hastie and Tibshirani (1993) is somewhat different than the one given by Eq. (1), in the way that the individual parameters in $\boldsymbol{\theta}(\cdot)$ depend on individual explanatory variables. In Anderson, Fang and Olkin (1994), the model given by Eq. (1) is denoted as a conditional parametric model, because when $\boldsymbol{x}_i$ is constant the model reduces to an ordinary linear model.

2.2. Local constant estimates

As only models where the parameters are functions of time are considered, only $\boldsymbol{x}_i = i$ is considered in the following. Estimation in Eq. (1) aims at estimating the functions $\boldsymbol{\theta}(\cdot)$, which in this case are the one-dimensional functions $\boldsymbol{\theta}(i)$. The functions are estimated only for distinct values of the argument t. Let t denote such a point and $\hat{\boldsymbol{\theta}}(t)$ the estimated coefficient functions, when the coefficients are evaluated at t.

One solution to the estimation problem is to replace $\boldsymbol{\theta}(i)$ in Eq. (1) with a constant vector $\boldsymbol{\theta}(i) = \boldsymbol{\theta}$ and fit the resulting model locally to t, using weighted least squares, i.e.

$$\hat{\boldsymbol{\theta}}(t) = \arg\min_{\theta} \sum_{i=1}^{t} w_i(t)(y_i - \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{\theta})^2. \tag{2}$$

Generally, using a nowhere increasing weight function $W: \boldsymbol{R}_0 \to \boldsymbol{R}_0$ and a spherical kernel the actual weight $w_i(t)$ allocated to the ith observation is determined by the Euclidean distance, in this case $|i - t|$, as

$$w_i(t) = W\left(\frac{|i-t|}{\hbar(t)}\right). \tag{3}$$

The scalar $\hbar(t)$ is called the bandwidth, and determines the size of the neighbourhood that is spanned by the weight function. If, e.g., $\hbar(t)$ is constant for all values of t it is denoted as a fixed bandwidth. In practice, however, also the nearest-neighbour bandwidth, which depends on the distribution of the explanatory variable, is used (Cleveland & Devlin, 1988). Although, in this case where $\boldsymbol{x}_i = i$, i.e. the distribution of the explanatory variable is rectangular, a fixed bandwidth and a nearest-neighbour bandwidth are equivalent.

2.3. Local polynomial estimation

If the bandwidth $\hbar(t)$ is sufficiently small the approximation of $\boldsymbol{\theta}(t)$ as a constant vector near t is good. This implies, however, that a relatively low number of observations is used to estimate $\boldsymbol{\theta}(t)$, resulting in a noisy estimate. On the contrary a large bias may appear if the bandwidth is large.

It is, however, obvious that locally to t the elements of $\boldsymbol{\theta}(t)$ may be better approximated by polynomials, and in many cases polynomials will provide good approximations for larger bandwidths than local constants. Local polynomial approximations are easily included in the method described. Let $\theta_j(t)$ be the jth element of $\boldsymbol{\theta}(t)$ and let $\boldsymbol{p}_d(t)$ be a column vector of terms in a d-order polynomial evaluated at t, i.e. $\boldsymbol{p}_d(t) = [t^d \;\; t^{d-1} \;\; \cdots \;\; 1]$. Furthermore, introduce $\boldsymbol{z}_i = [z_{1i} \;\; \cdots \;\; z_{pi}]^{\mathrm{T}}$,

$$\boldsymbol{u}_{i,t}^{\mathrm{T}} = [z_{1i}\boldsymbol{p}_{d_1}^{\mathrm{T}}(t-i) \cdots z_{ji}\boldsymbol{p}_{d_j}^{\mathrm{T}}(t-i) \cdots z_{pi}\boldsymbol{p}_{d_p}^{\mathrm{T}}(t-i)], \tag{4}$$

$$\hat{\boldsymbol{\phi}}^{\mathrm{T}}(t) = [\hat{\boldsymbol{\phi}}_1^{\mathrm{T}}(t) \cdots \hat{\boldsymbol{\phi}}_j^{\mathrm{T}}(t) \cdots \hat{\boldsymbol{\phi}}_p^{\mathrm{T}}(t)], \tag{5}$$

where $\hat{\boldsymbol{\phi}}_j(t)$ is a column vector of local constant estimates at t, i.e.

$$\hat{\boldsymbol{\phi}}_j^{\mathrm{T}}(t) = [\hat{\phi}_{jd_j+1}(t) \cdots \hat{\phi}_{j1}(t)] \tag{6}$$

corresponding to $z_{ji}\boldsymbol{p}_{d_j}^{\mathrm{T}}(t-i)$. Now weighted least-squares estimation is applied as described in Section 2.2, but fitting the linear model

$$y_i = \boldsymbol{u}_{i,t}^{\mathrm{T}}\boldsymbol{\phi} + e_i; \quad i = 1,\dots,t, \tag{7}$$

locally to t, i.e. the estimate $\hat{\boldsymbol{\phi}}(t)$ of the parameters $\boldsymbol{\phi}$ in Eq. (7) becomes a function of t as a consequence of the weighting. Estimates of the elements of $\boldsymbol{\theta}(t)$ can now be obtained as

$$\hat{\theta}_j(t) = \boldsymbol{p}_{d_j}^{\mathrm{T}}(0)\hat{\boldsymbol{\phi}}_j(t) = [\underbrace{0 \quad \cdots \quad 0 \quad 1}_{d_j+1}]\hat{\boldsymbol{\phi}}_j(t) = \hat{\phi}_{j1}(t); \quad j = 1,\dots,p. \tag{8}$$

3. Recursive least squares with forgetting factor

In this section the well-known RLS algorithm with forgetting factor is compared to the proposed method of estimation for the varying-coefficient approach. Furthermore, it is shown how to include local polynomial approximations in the RLS algorithm.

3.1. *The weight function*

The RLS algorithm with forgetting factor aims at estimating the parameters in the linear model

$$y_i = \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{\theta} + e_i \tag{9}$$

which corresponds to Eq. (1) when $\boldsymbol{\theta}(\boldsymbol{x}_i)$ is replaced by a constant vector $\boldsymbol{\theta}$. The parameter estimate $\hat{\boldsymbol{\theta}}(t)$, using the RLS algorithm with constant forgetting factor λ, is given by

$$\hat{\boldsymbol{\theta}}(t) = \arg\min_{\theta} \sum_{i=1}^{t} \lambda^{t-i}(y_i - \boldsymbol{z}_i^{\mathrm{T}}\boldsymbol{\theta})^2. \tag{10}$$

In this case the weight which is assigned to the ith observation in Eq. (10) can be written as

$$w_i(t) = \lambda^{t-i} = \left[\exp\left(\frac{i-t}{(\ln\lambda)^{-1}}\right)\right]^{-1} = \left[\exp\left(\frac{|i-t|}{-(\ln\lambda)^{-1}}\right)\right]^{-1}, \tag{11}$$

where the fact that $i \le t$ in Eq. (10) is used. Now it is easily seen that Eq. (11) corresponds to Eq. (3) with a fixed bandwidth $\hbar(t) = \hbar = -(\ln\lambda)^{-1}$, which furthermore shows how the bandwidth and the forgetting factor are related. By also comparing Eqs. (9) and (1) it is thus verified that the RLS algorithm with forgetting factor corresponds to local constant estimates in the varying-coefficient approach, with the specific choice Eq. (11) of the weight function.

3.2. *Recursive local polynomial approximation*

The RLS algorithm is given by Ljung and Soderstrom (1983)

$$\boldsymbol{R}(t) = \sum_{i=1}^{t} \lambda^{t-i}\boldsymbol{z}_t\boldsymbol{z}_t^{\mathrm{T}} = \lambda\boldsymbol{R}(t-1) + \boldsymbol{z}_t\boldsymbol{z}_t^{\mathrm{T}}, \tag{12}$$

$$\hat{\boldsymbol{\theta}}(t) = \hat{\boldsymbol{\theta}}(t-1) + \boldsymbol{R}^{-1}(t)\boldsymbol{z}_t[y_t - \boldsymbol{z}_t^{\mathrm{T}}\hat{\boldsymbol{\theta}}(t-1)] \tag{13}$$

with initial values

$$\boldsymbol{R}^{-1}(0) = \alpha\boldsymbol{I}, \quad \boldsymbol{\theta}(0) = \boldsymbol{0},$$

where α is large (Ljung & Soderstrom, 1983). Hence, the recursive algorithm is only asymptotically equivalent to solving the least-squares criteria, Eq. (10), which on the other hand does not give a unique solution for small values of t.

In Section 2.3 it was shown how to include local polynomial approximation of the parameters in the varying-coefficient approach, and that this could be done by fitting the linear model, Eq. (7), and calculating the parameters from Eq. (8). It is thus obvious to use the same approach in an extension of the RLS algorithm, replacing $\boldsymbol{z}_t$ by $\boldsymbol{u}_{i,t}$. However, the explanatory variable $\boldsymbol{u}_{i,t}$ is a function of t, which means that as we step forward in time,

$$\boldsymbol{R}(t-1) = \sum_{i=1}^{t-1} \lambda^{t-1-i}\boldsymbol{u}_{i,t-1}\boldsymbol{u}_{i,t-1}^{\mathrm{T}}$$

cannot be used in the updating formula for $\boldsymbol{R}(t)$, as $\boldsymbol{R}(t)$ depends on $\boldsymbol{u}_{i,t}$. To solve this problem a linear operator which is independent of t, and maps $\boldsymbol{p}_{d_j}(s)$ to $\boldsymbol{p}_{d_j}(s+1)$ has to be constructed. Using the coefficients of the relation

$$(s+1)^d = s^d + ds^{d-1} + \frac{d(d-1)}{2!}s^{d-2} + \cdots + 1. \tag{14}$$

It follows that

$$\boldsymbol{p}_{d_j}(s+1) = \begin{bmatrix} 1 & d_j & \dfrac{d_j(d_j-1)}{2!} & \dfrac{(d_j-1)(d_j-2)}{3!} & \cdots & 1 \\ 0 & 1 & d_j-1 & \dfrac{(d_j-1)(d_j-2)}{2!} & \cdots & 1 \\ & & 1 & d_j-2 & \cdots & 1 \\ & & & 1 & & \\ \vdots & & & & \ddots & \vdots \\ 0 & \cdots & & & \cdots & 1 \end{bmatrix} \begin{bmatrix} s^{d_j} \\ s^{d_j-1} \\ \vdots \\ 1 \end{bmatrix} = \boldsymbol{L}_j\boldsymbol{p}_{d_j}(s). \tag{15}$$

Since $\boldsymbol{L}_j$ is a linear operator it can be applied directly to $\boldsymbol{u}_{i,t} = \boldsymbol{L}\boldsymbol{u}_{i,t-1}$, where

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{L}_1 & 0 & 0 & 0 & 0 \\ 0 & \boldsymbol{L}_2 & 0 & 0 & 0 \\ \vdots & & & & \vdots \\ 0 & \cdots & & \cdots & \boldsymbol{L}_p \end{bmatrix} \tag{16}$$

which when applied to the recursive calculation Eq. (12) of $\boldsymbol{R}(t)$, yields

$$\boldsymbol{R}(t) = \lambda \boldsymbol{L}\boldsymbol{R}(t-1)\boldsymbol{L}^{\mathrm{T}} + \boldsymbol{u}_t\boldsymbol{u}_t^{\mathrm{T}} \tag{17}$$

and the updating formula for the parameters, Eq. (13), is left unchanged. The proposed algorithm will be denoted Polynomial RLS (POLRLS) in the following.

Note that if the polynomials in Eq. (4) were calculated for the argument i instead of $t - i$, then $\boldsymbol{u}_{i,t} = \boldsymbol{u}_{i,t-1}$, and it is seen that the recursive calculation in Eq. (12) could be used without modification, but now there would be a numerical problem for $t \to \infty$.

4. Simulation study

Simulation is used to compare the RLS and POLRLS algorithms. For this purpose we have generated $N = 11$ samples of $n = 1000$ observations from the time-varying ARX-model

$$y_i = ay_{i-1} + b(i)z_i + e_i, \quad e_i \in \mathrm{N}(0,1),$$

where

$$a = 0.7, \quad b(i) = 5 + 4\sin\left(\frac{2\pi}{1000}i\right), \quad z_i \in \mathrm{N}(0,1).$$

The estimation results are compared using the sample mean of the mean square error (*MSE*) of the deviation between the true and the estimated parameters:

$$MSE_a = \frac{1}{N-1}\sum_{j=2}^{N}\left\{\frac{1}{n-s+1}\sum_{i=s}^{n}(a - \hat{a}(i))^2\right\},$$

$$MSE_b = \frac{1}{N-1}\sum_{j=2}^{N}\left\{\frac{1}{n-s+1}\sum_{i=s}^{n}(b(i) - \hat{b}(i))^2\right\}$$

and the sample mean of the *MSE* of the predictions

$$MSE_p = \frac{1}{N-1}\sum_{j=2}^{N}\left\{\frac{1}{n-s+1}\sum_{i=s}^{n} (y_i - \hat{a}(i-1)y_{i-1} - \hat{b}(i-1)z_i)^2\right\}. \tag{18}$$

Only observations for which $i \geq s = 350 > \max(\hbar_{\mathrm{opt}})$, where $\hbar_{\mathrm{opt}}$ is the optimal bandwidth, are used in the calculation of the *MSE*, to make sure that the effect of the initialisation has almost vanished. The observations used for the prediction in Eq. (18), has not been used for the estimation of the parameters, therefore the optimal

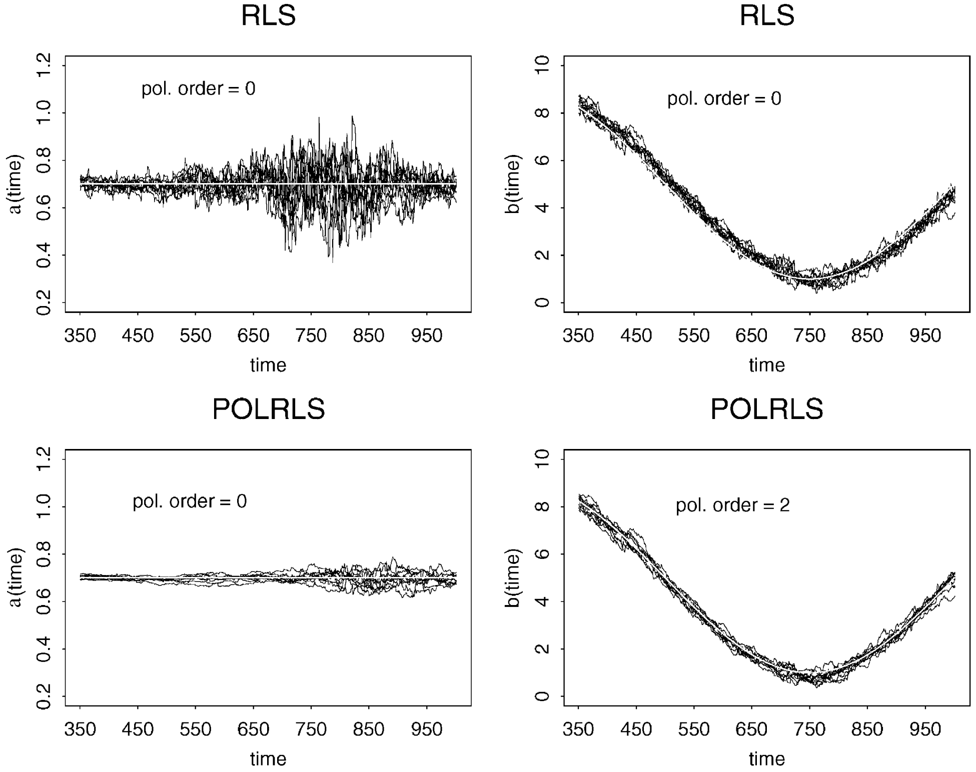

Fig. 1. Estimated parameter trajectories. The first row shows the trajectories from the RLS algorithm, the second row shows the result from the POLRLS algorithm where a has been approximated by a zero-order polynomial, and b by a second-order polynomial.

Table 1
MSE results using the RLS and POLRLS algorithms

Method	Pol. order	$\hbar_{opt}$	MSE_p	MSE_a	MSE_b
POLRLS	$d_1 = 2, d_2 = 2$	62	1.0847	0.0024	0.0605
POLRLS	$d_1 = 0, d_2 = 2$	57	1.0600	0.0005	0.0580
RLS	$d_1 = 0, d_2 = 0$	11	1.1548	0.0044	0.0871

bandwidth, $\hbar_{opt}$, can be found by minimising Eq. (18) with respect to the bandwidth $\hbar$, i.e. forward validation. The optimal bandwidth is found using the first sample, $j = 1$, the 10 following are used for the calculation of the sample means.

The POLRLS method was applied with two different sets of polynomial orders. The results are shown in Fig. 1 and Table 1. Obviously, knowing the true model, a zero-order polynomial approximation of a and a second-order polynomial approximation of b, should be the most adequate choice. In a true application such knowledge might not be available, i.e. if no preliminary analysis of data is performed. Therefore, a second-order polynomial approximation is used for both parameters, as this could be the default or standard choice. In both cases the POLRLS algorithm performs significantly better than the RLS algorithm, and, as expected, using a second-order approximation of a increases the *MSE* because in this case the estimation is disturbed by non-significant explanatory variables. In the figure it is seen, that it is especially when the value of $b(i)$ is small, that the variance of $\hat{a}$ is large. In this case the signal-to-noise ratio is low, and the fact that a larger bandwidth can be used in the new algorithm, means that the variance can be significantly reduced. Furthermore, it is seen that the reduction of the parameter estimation variance is greater for the fixed parameter than the time-varying parameter. The reason for this is that the optimal bandwidth is found by minimising the *MSE* of the predictions, and bias in the estimate of b contributes relatively more to the *MSE* than variance in the estimate of a, i.e. the optimal value of $\hbar$ balances bias in the estimate of b and variance in the estimate of a. When a second-order polynomial is used instead of a zero-order polynomial, for the estimation of b, it is possible to avoid bias even when a significantly larger bandwidth is used.

5. Summary

In this paper the similarity between the varying-coefficient approach and the RLS algorithm with forgetting factor has been demonstrated. Furthermore, an extension of the RLS algorithm, along the lines of the varying-coefficient approach is suggested. Using an example it is shown that the new algorithm leads to an significant improvement of the estimation performance, if the variation of the true parameters is smooth.

References

Anderson, T. W., Fang, K. T., & Olkin, I. (1994). Coplots, nonparametric regression, and conditionally parametric fits. *Multivariate analysis and its applications* (pp. 21–36). Hayward: Institute of Mathematical Statistics.

Cleveland, W. S., & Devlin, S. J. (1988). Locally weighted regression: An approach to regression analysis by local fitting. *Journal of the American Statistical Association*, *83*, 596–610.

Fan, J., Hardle, W., & Mammen, E. (1998). Direct estimation of low dimensional components in additive models. *The Annals of Statistics*, *26*, 943–971.

Hastie, T., & Tibshirani, R. (1990). *Generalized additive models*. London: Chapman & Hall.

Hastie, T., & Tibshirani, R. (1993). Varying-coefficient models. *Journal of the Royal Statistical Society B, Methodological*, *55*, 757–796.

Holt, C. (1957). Forecasting trends and sesonals by exponentially weighted moving averages. *O.N.R. Memorandum 52*. Carnegie Institute of Technology.

Ljung, L., & Soderstrom, T. (1983). *Theory and pratice of recursive indentification*. Cambridge, MA: MIT Press.

Nielsen, H. A., Nielsen, T. S., Madsen, H., & Joensen, A. (2000). Tracking time-varying coefficient-functions, *International Journal of Adaptive Control and Signal Processing*, accepted.

Parkum, J. E., Poulsen, N. K., & Holst, J. (1992). Recursive forgetting algorithms. *International Journal of Control*, *55*, 109–128.

Winters, P. (1960). Forecasting sales by exponentially weighted movings averages. *Management Sciences*, *6*, 324–342.

Alfred Joensen received the M.Sc. degree in electrical engineering from the Technical University of Denmark in 1997. In 1998/1999 he was employed as a Sr. research associate at the Department of Electrical Engineering, Texas Tech University. Currently, he is working at the Department of Mathematical Modelling, Technical University of Denmark and the Department of Wind Energy and Atmospheric Physics, Risø National Laboratory, in connection with his Ph.D. project. His research interests are in the field of applied statistics; in particular recursive estimation methods, non- and semi-parametric methods, and applications in boundary layer meteorology.

Henrik Madsen 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 Ass. Prof. in Statistics in 1986, Assoc. Prof. 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 organizations 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 Environ-metrics, Energy systems, Informatics and Finance. He has authored or co-authored approximately 190 papers and technical reports, and about 10 educational texts. He is in charge of three courses at the Technical University of Denmark.

Henrik Aa. Nielsen received the M.Sc. degree in engineering from the Technical University of Denmark in 1991. He worked as statistician in the pharmaceutical industry from 1991 to 1994. Hereafter he has been with the Department of Mathematical Modelling, Technical University of Denmark, where he has been employed as research assistant, research fellow, and assistant research professor. His research interests are within the field of applied statistics; mainly non- and semi-parametric methods with applications to energy systems.

Torben Skob Nielsen received the M.Eng.Sc. degree in control engineering from the Technical University of Denmark in 1990. Hereafter he spent 3 years as a project engineer at LicEnergy, working in the area of flow modelling in pipeline systems. He then took up a position as research assistance at the Department of Mathematical Modelling, Technical University of Denmark. In 1996 he became a research fellow and in 1999 he took up a position as Assistent Research Professor at the same department. His interests are in the area of prediction and control of non-linear stocastic systems — primarly within the energy sector.