Proceedings of the 40th IEEE
Conference on Decision and Control
Orlando, Florida USA, December 2001

Identification of a Weighted Combination of Multivariable Local Linear State-Space Systems from Input and Output Data

Vincent Verdult and Michel Verhaegen

Faculty of Applied Physics, University of Twente
P. O. Box 217, NL-7500 AE Enschede, The Netherlands
V.Verdult@tn.utwente.nl, M.Verhaegen@tn.utwente.nl

Abstract

The paper discusses a method for the determination of a weighted combination of local linear state-space systems from input and output data. The method is iterative and each iteration consists of two steps. The first step is to determine the weighting functions given the local models. This problem is solved by using an extended Kalman smoother. The second step is to identify the local models given the weights. For this step we optimize a cost function that represents a trade-off between local and global learning. For this optimization we use a gradient search method in combination with an appropriate projection in the parameter space to deal with similarity transformations.

1 Introduction

For the identification of nonlinear systems several approaches have been proposed [1]. One popular approach is the identification of a weighted combination of local linear models [2]. Methods exist for both input-output models and state-space models. In this paper we propose a new method for the identification of a weighted combination of local linear state-space models. The method identifies the local linear models as well as the time-varying weights. It can deal with multivariable input and output data and with the case where no state measurements are available. Basically, the method is a nonparametric one, because there is no need to parameterize the weighting functions, and no need to choose a specific parameterization for the local linear models; the local linear models are fully parameterized and the active parameters are determined numerically. Most existing methods require the parameterization of the weighting functions. It can however be beneficial to have a nonparametric approach like the one we present in this paper. One reason is that this makes it possible to estimate the weights first and based on these results determine an appropriate parameterization and optionally re-estimate the parameters for this parameterization. Another reason is that in this way we can deal with weighting functions that have nonconnected support. In the conventional methods, this would lead to the introduction of a new local model for each connected part of the weights.

Section 2 introduces the model structure we consider, and explains the basic idea of the identification method. Section 3 describes a method to estimate the weighting functions using an extended Kalman smoother. Section 4 discusses the identification of the local linear models. A simple numerical example is provided in Section 5.

2 The Basic Idea

Consider the model structure

$$\begin{aligned} x(k+1) &= \sum_{i=1}^{M} p_i(k)\Big(A_i x(k) + B_i u(k)\Big) \\ y(k) &= Cx(k) + Du(k), \end{aligned}$$

where $p_i(k)$ are the time-varying weights that determine which model is active, and M is the total number of models. The weights $p_i(k)$ are required to satisfy at each time instant k

$$0 \leq p_i(k) \leq 1, \quad \sum_{i=1}^{M} p_i(k) = 1. \qquad (1)$$

For ease of notation later on, we introduce the vector $p(k)$ as $p(k) = [p_1(k)^T, \ldots, p_M(k)^T]^T$.

0-7803-7061-9/01/$10.00 © 2001 IEEE

Offsets on the state equation can easily be incorporated by extending the input with a one and redefining the B_i and D matrices.

It is assumed that only measurements of $u(k)$ and $y(k)$ are available. The task is to determine the weights $p(k)$ and the system matrices. Given an initial guess of the system matrices A_i, B_i, C, D of the M local models, the identification procedure is an iterative one consisting of two steps: 1) Estimate the weights given the system matrices; 2) Estimate the system matrices given the weights. These two steps are discussed below. At present we do not know much about the interaction of these two steps, let alone the convergence of the scheme. Note that the two problems addressed by the above listed steps are also of interest on their own. The estimation of weights given the system matrices is a problem that often occurs in developing control schemes for local linear models [3]. The estimation of the local models given the weights is an identification problem for a linear parameter-varying system that is affine in the parameters and for which these parameters are known a priori.

3 Estimation of the Weights

This section addresses the problem of estimating the weights $p(k)$ given the system matrices. This task is complicated by the fact that the state sequence $x(k)$ is unknown. We propose to treat the weights $p(k)$ as additional states and use an extended Kalman smoother to estimate them. Without any prior information on the weights $p(k)$ we pose the following dynamic relation for them: $p(k+1) = p(k) + \xi(k)$ where $\xi(k)$ is a fictitious noise sequence. By controlling the variance of $\xi(k)$, the rate of change of $p(k)$ is controlled. This means that the covariance matrix of $\xi(k)$ that is used in the Kalman filter controls the smoothness of the weights $p(k)$. A similar, but different approach to estimate the weights using Kalman filtering has been described in [4].

Treating the weights as an additional state we get the following extended system

$$\begin{bmatrix} x(k+1) \\ p(k+1) \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{M} p_i(k)\Big(A_i x(k) + B_i u(k)\Big) \\ p(k) + \xi(k) \end{bmatrix} \quad (2)$$

$$y(k) = Cx(k) + Du(k). \quad (3)$$

For ease of notation we introduce the extended state $z(k) = [x(k)^T, p(k)^T]^T$ and write the extended system as follows:

$$z(k+1) = f\Big(z(k), u(k)\Big) + w(k) \quad (4)$$

$$y(k) = C_e z(k) + Du(k) + v(k), \quad (5)$$

with the obvious definition of the function $f(\cdot)$ and the matrix C_e. The signal $w(k)$ represents the sequence $\xi(k)$ together with possible process noise on $x(k)$. The signal $v(k)$ represents the measurement noise. It is assumed that the signals $v(k)$ and $w(k)$ are statistically independent white noise sequences. The covariance matrices for $w(k)$ and $v(k)$ are denoted by Q and R, respectively. Let the $\hat{z}(k|k-1)$ denote the estimate of the extended state at time instant k given the measurement up to time instant $k-1$, let the square root of its covariance matrix be $S(k|k-1)$. Then with the generalized covariance representation we can write

$$\hat{z}(k|k-1) = z(k) + S(k|k-1)\epsilon_1(k) \quad (6)$$

$$y(k) = C_e z(k) + Du(k) + R^{\frac{1}{2}}\epsilon_2(k) \quad (7)$$

$$z(k+1) = f\Big(z(k), u(k)\Big) + Q^{\frac{1}{2}}\epsilon_3(k). \quad (8)$$

From the work of Duncan and Horn [5] we know that the minimum variance estimate of the state can be obtained by minimizing $\epsilon(k)^T\epsilon(k)$, $\epsilon(k) = [\epsilon_1(k)^T, \epsilon_2(k)^T, \epsilon_3(k)^T]^T$ with respect to $z(k)$ and $z(k+1)$ subject to the constraints (6)–(8). This is a nonlinear optimization problem. We use an iterative approach to solve it. We linearize the function $f(\cdot)$ around the vector $z_i(k)$;

$$f\Big(z(k), u(k)\Big) \approx f\Big(z_i(k), u(k)\Big) + F\Big(z_i(k), u(k)\Big)\Big(z(k) - z_i(k)\Big),$$

where $F(z_i(k), u(k))$ is the Jacobian of $f(\cdot)$ evaluated at $z_i(k)$. With this approximation we arrive at a linear least squares problem; we minimize

$$\left\| W(k)^{-1}\Big(\Phi_i(k)\begin{bmatrix} z(k) \\ z(k+1) \end{bmatrix} - \zeta_i(k)\Big)\right\|_2^2, \quad (9)$$

where

$$W(k) = \begin{bmatrix} S(k|k-1) & 0 & 0 \\ 0 & R^{\frac{1}{2}} & 0 \\ 0 & 0 & Q^{\frac{1}{2}} \end{bmatrix}$$

$$\Phi_i(k) = \begin{bmatrix} I_{n+M} & 0 \\ C_e & 0 \\ F\Big(z_i(k), u(k)\Big) & -I_{n+M} \end{bmatrix}$$

$$\zeta_i(k) = \begin{bmatrix} \widehat{z}(k|k-1) \\ y(k) - Du(k) \\ -f\Big(z_i(k),u(k)\Big) + F\Big(z_i(k),u(k)\Big)z_i(k) \end{bmatrix}.$$

We use a QR factorization to solve this problem:

$$W(k)^{-1}\Phi_i(k) = T_i(k)\begin{bmatrix} \overline{R}_i(k) & \overline{G}_i(k) \\ 0 & S_i(k+1|k)^{-1} \\ 0 & 0 \end{bmatrix},$$

where $T_i(k)$ is an orthogonal matrix. Minimizing equation (9) is equivalent to minimizing

$$\left\| \begin{bmatrix} \overline{R}_i(k) & \overline{G}_i(k) \\ 0 & S_i(k+1|k)^{-1} \end{bmatrix} \begin{bmatrix} z(k) \\ z(k+1) \end{bmatrix} - \begin{bmatrix} \overline{c}_i(k) \\ c_i(k+1) \end{bmatrix} \right\|_2^2 \tag{10}$$

with

$$\begin{aligned} \overline{c}_i(k) &= [I_{n+M}, 0_{(n+M)\times(n+M+\ell)}] \\ &\quad \times T_i(k)^T W(k)^{-1}\zeta_i(k) \\ c_i(k+1) &= [0_{(n+M)\times(n+M)}, I_{n+M}, 0_{(n+M)\times\ell}] \\ &\quad \times T_i(k)^T W(k)^{-1}\zeta_i(k). \end{aligned}$$

The solution to this set of equations only depends on measurements up to time instant k, therefore it equals the one-step ahead predicted state

$$\widehat{z}_i(k+1|k) = S_i(k+1|k)c_i(k+1),$$

and the filtered state estimate

$$\widehat{z}_i(k|k) = \overline{R}_i(k)^{-1}\Big(\overline{c}_i(k) - \overline{G}_i(k)\widehat{z}_i(k+1|k)\Big).$$

Let $z_i(k)$ for $i = 0$ be equal to $\widehat{z}(k|k-1)$. We minimize (10) with respect to $z(k)$ and $z(k+1)$. This gives us the new estimates $\widehat{z}_i(k|k)$ and $\widehat{z}_i(k+1|k)$. The solution $\widehat{z}_i(k|k)$ is used to improve the linearization of the function $f(\cdot)$; we take $z_{i+1}(k) = \widehat{z}_i(k|k)$ and minimize (10) again. We continue this iteration until we have convergence established by the rule $||z_i(k|k) - z_{i-1}(k|k)||_2^2 < \varepsilon$. Alternatively, we could simply use a fixed number of iterations. After completing this iteration we obtain the final estimates $\widehat{z}(k|k)$ and $\widehat{z}(k+1|k)$; and the corresponding matrices $\overline{R}(k)$, $\overline{G}(k)$, $S(k+1|k)$, and $\overline{c}(k)$. To increase the robustness of this scheme we apply regularization if the condition number of the matrix $\Phi_i(k)$ becomes too large.

With the solution $\widehat{z}(k+1|k)$ and its covariance $S(k+1|k)$, we can construct the matrices $W(k+1)$, $\Phi_i(k+1)$, and $\zeta_i(k+1)$ and compute the state estimate $\widehat{z}(k+1|k+1)$. This way we can compute the predicted state estimate $\widehat{z}(k+1|k)$ and the filtered state estimate $\widehat{z}(k|k)$ at the time instances $k = 0, 1, \ldots, N-1$.

From equation (10) it is easy to see that for the time window $k = 0, 1, \ldots, N-1$ we get the following set of equations:

$$\begin{bmatrix} \overline{R}(0) & \overline{G}(0) & & \\ & \overline{R}(1) & \overline{G}(1) & \\ & & \ddots & \ddots \\ & & & \overline{R}(N-1) \end{bmatrix} \begin{bmatrix} z(0) \\ z(1) \\ \vdots \\ z(N-1) \end{bmatrix} \approx \begin{bmatrix} \overline{c}(0) \\ \overline{c}(1) \\ \vdots \\ \overline{c}(N-1) \end{bmatrix}$$

with $\overline{R}(N-1) = S(N|N-1)$ and $\overline{c}(N-1) = c(N)$. These equations show that the smoothed state sequence $\widehat{z}(k|N-1)$, $k = 0, 1, \ldots, N-1$ can be estimated using the $N-1$ available measurements by back substitution as follows

$$\widehat{z}(k|N-1) = \overline{R}(k)^{-1}\Big(\overline{c}(k) - \overline{G}(k)\widehat{z}(k+1|N-1)\Big), \tag{11}$$

with $\widehat{z}(N-1|N-1) = R(N-1)^{-1}\overline{c}(N-1)$.

We want the estimates of the weighting functions $p(k)$ to satisfy the constraints (1). These constraints can easily be incorporated in the extended Kalman smoother in a point-wise fashion. Instead of computing $\widehat{z}(k|N-1)$ (and thus $\widehat{p}(k|N-1)$) with equation (11) we compute them as the minimizers $z(k)$ of

$$\left\| \overline{R}(k)z(k) - \Big(\overline{c}(k) - \overline{G}(k)\widehat{z}(k+1|N-1)\Big) \right\|_2^2,$$

subject to $\begin{bmatrix} 0 & e^T \end{bmatrix} z(k) = 1$, and $\begin{bmatrix} 0 & I_M \end{bmatrix} z(k) \geq 0$, where e is an M-dimensional vector of ones. This is a quadratic programming problem for which efficient numerical solvers exist [6]. The starting value $\widehat{z}(N-1|N-1)$ for this recursion is obtained by minimizing

$$\left\| \overline{R}(N-1)z(N-1) - \overline{c}(N-1) \right\|_2^2$$

subject to similar constraints.

4 Estimation of the Local Models

Since the extended Kalman smoother described in the previous section, estimates both the state and the weights, one could be tempted to use these estimates and solve a linear least squares problem to obtain the system matrices. However, this is not an appropriate method, because the system matrices that form the solution to this problem are exactly those that where used in the extended Kalman smoother. Therefore, this does not lead to an update of the system matrices. To update the system matrices, we should not use the state estimate of the extended Kalman smoother, but

only the weight estimate. This unfortunately complicates the identification procedure, the resulting problem is the identification of a linear parameter-varying system.

Let $y(k)$ represent the measurement of the output of the underlying system and let $\widehat{y}(k,\theta)$ represent the output of the model parameterized by θ. To fit the model to the data, we want to minimize the following cost function:

$$E_N(\theta) := \sum_{k=0}^{N-1} ||y(k) - \widehat{y}(k,\theta)||_2^2.$$

This basically means that we adopt a global learning strategy. However, it has been argued that for problems in which the model structure does not match the underlying system, or the complexity in the input-output data, global learning does not give satisfactory generalization results [7], [8]. That means that the estimated models perform poorly on data that were not used for training. Therefore, in addition to a good global fit to the output of the underlying system, local linear models are often required to also approximate the local linearizations of the underlying system. For a further discussion on this topic we refer the reader to [8]. We introduce the local state $x_i(k)$ and local output $y_i(k)$ as follows:

$$\begin{aligned} x_i(k+1) &= p_i(k)\Big(A_i x_i(k) + B_i u(k)\Big) \\ y_i(k) &= C x_i(k), \end{aligned}$$

and we define local cost functions

$$E_N^i(\theta_i) := \sum_{k=0}^{N-1} \Big\| \rho_i(k)\Big(y(k) - \widehat{y}_i(k,\theta_i)\Big)\Big\|_2^2,$$

where $\rho_i(k)$ is a weighting function with local support for the i-th model; one possible choice is $\rho_i(k) = p_i(k)$. Combining the global and local cost functions we arrive at

$$V_N(\theta) := E_N(\theta) + \lambda \sum_{i=1}^{M} E_N^i(\theta_i),$$

where λ has to be chosen by the user and determines the trade-off between local and global learning. Note that in general minimizing this combined cost function results in a nonlinear, nonconvex optimization problem.

We propose to use an iterative Newton-type method to minimize the cost function $V_N(\theta)$. Such a method updates the system parameters θ with the following rule:

$$\theta^{(i+1)} = \theta^{(i)} - \mu^{(i)} V_N''(\theta^{(i)})^{-1} V_N'(\theta^{(i)}), \quad (12)$$

where $V_N'(\theta)$ is the Jacobian of $V_N(\theta)$, $V_N''(\theta)$ is the Hessian of $V_N(\theta)$, and $\mu^{(i)}$ is a step size that can for example be determined by numerical interpolation.

To minimize the cost function $V_N(\theta)$, a parameterization θ of the local linear models is needed. Instead of using local canonical forms, the matrices A, B, C, and D, are fully parameterized, and the active parameters, that is, the parameters that change the cost function, are determined numerically at each iteration of the nonlinear optimization procedure. It is important to realize that the minimization of $V_N(\theta)$ with a full parameterization does not yield a unique solution, because there exist different systems that have the same input-output behavior (simply apply a similarity transformation to the state). To avoid problems during the iterative minimization, not all parameters, but only the parameters that change the cost function are updated. Such an approach has been described by Lee and Poolla [9] for the identification of linear parameter-varying systems. The advantage of such an approach is that there is no need for a special parameterization of the system. This approach avoids ill-conditioning that often occurs with canonical parameterizations. What follows is a brief description how the method of Lee and Poolla can be used for the identification of local linear models.

To facilitate the exposition, we formally define the system parameters θ as $\theta = P\,\mathrm{vec}(\Theta)$ with Θ given by

$$\begin{aligned} \Theta &:= \left[\begin{array}{cc} \overline{A} & \overline{B} \\ \overline{C} & \overline{D} \end{array}\right] \\ &:= \left[\begin{array}{ccccccc} 0 & A_1 & \cdots & A_M & 0 & B_1 & \cdots & B_M \\ C & 0 & \cdots & 0 & D & 0 & \cdots & 0 \end{array}\right], \end{aligned}$$

and P is a selection matrix that discards the entries in Θ that are zero by definition.

Similar to [9] we can define the similarity map $S_\Theta : T, \det(T) \neq 0 \rightarrow \mathbb{R}^{(n+\ell)\times(n+m)(M+1)}$ as:

$$S_\Theta(T) := \left[\begin{array}{cc} T^{-1} & 0 \\ 0 & I \end{array}\right] \left[\begin{array}{cc} \overline{A} & \overline{B} \\ \overline{C} & \overline{D} \end{array}\right] \left[\begin{array}{cc} T_{M+1} & 0 \\ 0 & I \end{array}\right],$$

where T_{M+1} denotes a block diagonal matrix with $M+1$ copies of the matrix T along the diagonal. Taking a certain model Θ, all models with the same input-output behavior are given by:

$$\mathcal{I}_\Theta := \left\{\overline{\Theta} \mid \overline{\Theta} = S_\Theta(T),\ \det(T) \neq 0\right\}.$$

This set is called the indistinguishable set at Θ, because it contains all the models that cannot be

distinguished from Θ by looking at their input-output behavior. According to Lee and Poolla [9] if the similarity map S_Θ is locally one-to-one around I_n, the connected component

$$\mathcal{I}_\Theta^+ := \{\overline{\Theta} \mid \overline{\Theta} = S_\Theta(T),\ \det(T) > 0\}$$

of $\mathcal{I}_\Theta$ is a manifold. The tangent space of this manifold at the point Θ contains the directions along which a change in the parameters θ does not influence the value of the cost function $V_N(\theta)$. Hence, it makes no sense changing the parameters into these directions during an update with (12). The idea is to modify the update rule (12) such that these directions are projected out of the search direction. Therefore, at each point $\theta^{(i)}$ we need to determine the tangent space of the manifold $\mathcal{I}_\Theta^+$. A derivation similar to the one presented in [9] shows that the tangent space at the point Θ equals the column space of the matrix

$$M_\Theta = \sum_{i=1}^{M+1} \begin{bmatrix} \Pi_i^T \\ 0_{m(M+1)\times n} \end{bmatrix} \otimes \begin{bmatrix} \overline{A}\Pi_i^T \\ \overline{C}\Pi_i^T \end{bmatrix} - \begin{bmatrix} \overline{A}^T \\ \overline{B}^T \end{bmatrix} \otimes \begin{bmatrix} I_n \\ 0_{\ell\times n} \end{bmatrix},$$

where $\Pi_i := [0_{n\times(i-1)n}, I_n, 0_{n\times(M+1-i)n}]$. Therefore, the left null space of the matrix M_Θ, which hence is the orthogonal complement of the tangent space of the manifold $\mathcal{I}_\Theta^+$ at the point Θ, contains the directions in which the parameters should be changed to obtain a change in the cost function $V_N(\theta)$. Taking into account the zeros in the matrix Θ, the directions in which to change the parameters can be determined from a singular value decomposition of PM_Θ:

$$PM_\Theta = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} V^T.$$

Now every parameter vector θ can be decomposed into two parts: $\theta = U_1U_1^T\theta + U_2U_2^T\theta$, where the first part corresponds to the directions that do not influence the cost function, and the second part to the directions that change the value of the cost function. Based upon this observation, the update rule for the parameters (12) is changed such that the update is restricted to the directions that change the cost function as follows:

$$\theta^{(i+1)} = \theta^{(i)} - \mu^{(i)}U_2\left(U_2^TV_N''(\theta^{(i)})U_2\right)^{-1} \times U_2^TV_N'(\theta^{(i)}).$$

5 Example

In this section we use a simple example to show the potential of the proposed method. The underlying system is in the model class. It consists of two single-input, single-output local linear models, given by:

$$A_1 = \begin{bmatrix} 0 & 0.8 \\ -0.8 & 0.5 \end{bmatrix},\quad B_1 = \begin{bmatrix} 0.4 \\ 0 \end{bmatrix}$$
$$A_2 = \begin{bmatrix} 0 & 0.5 \\ -0.5 & 0 \end{bmatrix},\quad B_2 = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}$$
$$C = \begin{bmatrix} 1 & 1 \end{bmatrix}.$$

The first local model has an offset on the state equal to $[0.5,\ 0]^T$, and the second local model $[0.2,\ 2]^T$. The weights are given by

$$p_1(k) = \frac{1}{2}\sin\left(\frac{2\pi}{500}k\right) + \frac{1}{2},\ p_2(k) = 1 - p_1(k).$$

The input signal is a zero mean white noise sequence. The output is perturbed by measurement noise with a variance of 0.01. Figure 1 shows the simulated output signal. For the Kalman smoother we take

$$Q = \begin{bmatrix} 10^{-6}I_n & 0 \\ 0 & 10^{-8}I_M \end{bmatrix},\ R = 10^{-6}.$$

In the cost function $V_N(\theta)$ we take $\lambda = 1$. The quality of the output signals generated by the identified model, is measured using the variance accounted for (VAF), which is defined as

$$\text{VAF} = \max\left\{1 - \frac{\text{var}(y_k - \widehat{y}_k)}{\text{var}(y_k)}, 0\right\} \times 100\%,$$

where $\widehat{y}_k$ denotes the estimated output signal, and $\text{var}(\cdot)$ denotes the variance of a quasi-stationary signal.

To start the iterative identification procedure, we need initial estimates of the local models. We generate two initial estimates by dividing the available data into two parts and estimating a linear model for each part using a subspace identification method [10]. This of course does not lead to very good initial estimates. In practice we can often make some educated guesses of how to divide the data in order to obtain some reasonable initial estimates of the local models. The table below lists the VAF values after each iteration. Note that one iteration consists of estimating weights followed by estimating the models.

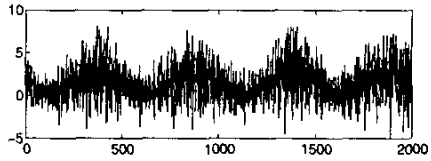

Figure 1: Output of the example system.

Iteration	1	2	3	4	5
VAF	79.73	98.02	99.47	99.64	99.66

After only five iterations, we already have a good model. Figure 2 shows the original time-varying weights, the estimates of these weights after the first iteration and the estimates after the fifth iteration. On a fresh data set, generated by taking a different realization of the input signal $u(k)$, the VAF of this model is 99.73%. We have not included a figure showing the output of the model, because it resembles the original output so closely that it is not possible to distinguish them on sight.

We have performed more simulations with the method that we propose in this paper. Here we have observed that in some simulations the non-convexity of the optimization problem resulted in a convergence to a local minimum. Therefore, the key element for further research is to supply the proposed method with good initial estimates of the local models.

6 Conclusion

We proposed an iterative method for the determination of a weighted combination of local linear state-space systems from input and output data. The weighting functions are obtained using an extended Kalman smoother. The local linear models are estimated by optimizing a cost function that represents the trade-off between local and global learning. Because of the preliminary state of research, the method that we proposed in this paper has only been evaluated on a number of experiments. These experiments show the potential of the method and motivate further research. To improve the method, further research will be devoted to the interaction between the estimation of the weights and the estimation of the models and to the generation of good initial estimates of the local models.

Acknowledgement

This work was supported by the Dutch Technology Foundation (STW) under project number DEL55.3891.

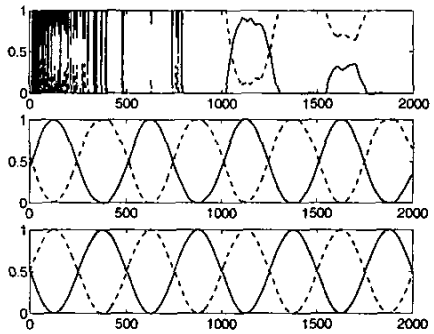

Figure 2: From top to bottom: The estimates of the time-varying weights after the first iteration, the estimates after the fifth iteration and the original time-varying weights.

References

[1] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.-Y. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modeling in system identification: A unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995.

[2] R. Murray-Smith and T. A. Johansen, *Multiple Model Approaches to Modelling and Control*. London: Taylor & Francis, 1997.

[3] K. S. Narendra, J. Balkrishnan, and M. K. Ciliz, "Adaptation and learning using multiple models, switching and tuning," *IEEE Control Systems Magazine*, pp. 37–51, June 1995.

[4] A. Banerjee, Y. Arkun, B. Ogunnaike, and R. Pearson, "Estimation of nonlinear systems using linear multiple models," *AIChE Journal*, vol. 43, pp. 1204–1226, May 1997.

[5] D. Duncan and S. Horn, "Linear dynamic recursive estimation from the viewpoint of regression analysis," *Journal of the American Statistical Association*, vol. 67, pp. 815–821, 1972.

[6] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*. London, UK: Academic Press, 1981.

[7] L. Bottou and V. Vapnik, "Local learning algorithms," *Neural Computation*, vol. 4, no. 6, pp. 888–900, 1992.

[8] R. Murray-Smith and T. A. Johansen, "Local learning in local model networks," in [2].

[9] L. H. Lee and K. Poolla, "Identifiability issues for parameter-varying and multidimensional linear systems," in *ASME Design Engineering Technical Conferences*, (Sacramento, California), Sept. 1997.

[10] M. Verhaegen, "Identification of the deterministic part of MIMO state space models given in innovations form from input-output data," *Automatica*, vol. 30, no. 1, pp. 61–74, 1994.