

Subspace identification of switching model

Komi Midzodzi PEKPE, Komi GASSO[#], Gilles MOUROT, José RAGOT
 Centre de Recherche en Automatique de Nancy-CNRS UMR 7039
 2, Avenue de la Forêt de Haye 54516 Vandoeuvre-lès-Nancy Cedex France
 Tel: (33) 3 83 59 57 04, Fax: (33) 3 83 59 56 44
 E-mail: {kpekpe, gmourot, jragot}@ensem.inpl-nancy.fr
[#]Laboratoire PSI - CNRS FRE N° 2645
 Place Emile Blondel - BP 08 F-76131 Mont-Saint-Aignan Cedex
 Tel : (33) 2 32 95 98 75, Fax: (33) 2 32 95 97 08
 E-mail: komi.gasso@insa-rouen.fr

Abstract— Subspace identification of switching model is considered in this paper. Here the switching model is supposed to be a sum of weighted linear models. The method established uses recursive subspace identification to estimate the switching function and least squares method for local model Markov parameters estimation. To perform the computation of the weighting functions a two-steps algorithm (switching times determination and model merging) is given. Finally the local model parameter estimation is based on the estimation of the Markov parameters.

Keywords— switching model, Markov parameters, least squares, weighting function, recursive subspace identification.

I. I

The objective of this paper is to identify switching models. Here a switching model structure is considered as a sum of weighted linear systems (local models), but only one model is active at any time. In this paper, we use a weighting function which determines the active model. Our goal is to estimate the weighting function and identify the local model, using the knowledge of the inputs and outputs only. Thus, knowing the weighting function allows to determine which local model is active. The estimation of the switching times is made by a recursive subspace identification method. The proposed method is not sensible to the change in input dynamics because the recursive subspace method proposed in [4] is not sensible to this fact. Then the switching times correspond only to a change in the system dynamic. To identify the local models, we use least squares method. The parameter estimation is done by the extraction of the Markov parameter matrix. The advantage of the proposed method results in fact that there is not any stage of nonlinear optimization, and the parameter estimation is unbiased.

After the formulation of the problem, the notations are introduced and the estimation of the Markov parameters is achieved. In section 4 the determination of the weighting function is performed. The identification of the local models procedure is given. Finally an example illustrates the performance of the proposed method.

II. F

The output of the switching model can be modeled as a weighted sum of outputs of h linear models as follows:

$$y_k = \sum_{s=1}^h \omega_{s,k} y_{s,k}, \quad (1)$$

where $y_k \in \mathbb{R}^\ell$, any weight $\omega_{s,k} \in [0, 1]$ and $\sum_{s=1}^h \omega_{s,k} = 1$, $k \in [1, q]$ (if we have q measurements of the inputs and outputs). $\omega_{s,k}$ represents the weighting function and $y_{s,k}$ the output of the s^{th} local model.

Any local model is supposed linear of order n_s and can be described by the equation:

$$\begin{aligned} x_{s,k+1} &= A_s x_{s,k} + B_s u_k, \\ y_{s,k} &= C_s x_{s,k} + D_s u_k + e_k \end{aligned} \quad (2)$$

Here the output error $e_k \in \mathbb{R}^\ell$ is assumed to be a zero mean white noise sequence and uncorrelated with $u_k \in \mathbb{R}^m$ and has covariance matrix

$$\mathbf{E} e_k e_t^T = \begin{cases} R > 0, & k=t \\ 0, & \text{otherwise.} \end{cases}$$

We suppose each local model is stable.

The inputs u_k and outputs y_k of the switching model are supposed to be known. The object is to determine:

- the weighting function $\omega_{s,k}$ for each model,
- the order n_s and the h linear local model parameters.

The parameters to be determined are A_s , B_s , C_s and D_s . Note that, to obtain the g^{th} weighting outputs, it is sufficient to make the product of the global outputs y_k by the g^{th} weighting function:

$$\omega_{g,k} y_k = \sum_{s=1}^h \omega_{g,k} \omega_{s,k} y_{s,k} = \omega_{g,k} y_{g,k}, \quad k \in [1, q] \quad (3)$$

because

$$(\omega_{g,k})^2 = \omega_{g,k} \text{ and } \omega_{g,k} \times \omega_{s,k} = 0, \text{ (if } g \neq s), \quad k \in [1, q] \quad (4)$$

In the following we use a recursive subspace identification technique to obtain the weighting function $\omega_{s,k}$. In order to obtain local Markov parameters, least squares tools will be used. In the next section the matrices used later are defined.

III. T

For the s^{th} model, we can set the following definitions (note that, for simplicity we use i instead of i_s in this article).

Output weighted block Hankel matrix of the s^{th} model is defined as ($i > n_s$):

$$Y_{s,\omega}^i = \begin{pmatrix} \omega_{s,i} y_{s,i} & \omega_{s,i+1} y_{s,i+1} & \dots & \omega_{s,i+j-1} y_{s,i+j-1} \end{pmatrix} \quad (5)$$

Input block Hankel matrix is defined as:

$$U = \begin{pmatrix} u_1 & u_2 & \dots & u_j \\ u_2 & u_3 & \dots & u_{j+1} \\ \vdots & \vdots & \ddots & \vdots \\ u_i & u_{i+1} & \dots & u_{i+j-1} \end{pmatrix} \quad (6)$$

The same definition is made for the Hankel block matrix E for the noise e_k .

The state sequence matrix X_s and the weighting matrix of the s^{th} model Ω_s are defined as:

$$X_s = \begin{pmatrix} x_{s,1}^T & x_{s,2}^T & \dots & x_{s,j}^T \end{pmatrix}^T \quad (7)$$

and

$$\Omega_s = \begin{pmatrix} \omega_{s,i} & 0 & 0 & 0 \\ 0 & \omega_{s,i+1} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \omega_{s,i+j-1} \end{pmatrix} \quad (8)$$

Then we have:

$$Y_{s,\omega}^i = \begin{pmatrix} y_{s,i} & y_{s,i+1} & \dots & y_{s,i+j-1} \end{pmatrix} \Omega_s \quad (9)$$

Y_s^i

The extended observability matrix $\Gamma_{i,s}$ of the s^{th} model is defined as:

$$\Gamma_{i,s} = \begin{pmatrix} C_s \\ C_s A_s \\ \vdots \\ C_s A_s^{i-1} \end{pmatrix} \mathbb{R}^{\ell_i \times n_s} \quad (10)$$

The extended controllability matrix $\mathcal{C}_{i,s}$ of the s^{th} model is defined as :

$$\mathcal{C}_{i,s} = \begin{pmatrix} B_s & A_s B_s & \dots & A_s^{i-1} B_s \end{pmatrix} \quad (11)$$

The Markov parameter matrix for the deterministic part H_i^d and for the stochastic part H_i^{st} of the system are defined

as¹:

$$H_{s,i}^d = \begin{pmatrix} C_s A_s^{i-2} B_s & C_s A_s^{i-3} B_s & \dots & C_s B_s & D_s \end{pmatrix} \quad (12)$$

$$H_{s,i}^{st} = \begin{pmatrix} 0 & 0 & \dots & 0 & I \end{pmatrix} \quad (13)$$

IV. E

M

Local matrix input-output equation

It is essential to write the matrix input-output relation, that allows the extraction of the Markov parameter matrix. The following theorem summarizes this relation.

Theorem 1:

$$Y_{s,\omega}^i = C_s A_s^{i-1} X_s \Omega_s + H_{s,i}^d U \Omega_s + H_{s,i}^{st} E \Omega_s \quad (14)$$

The proof of the theorem is established in appendix 1.

Now, the objective is to eliminate the term depending on the state X_s and the noise E , in order to obtain the Toeplitz matrix

$H_{s,i}^d$. The proposed method uses least squares and may be summarize in the next theorem.

Theorem 2: Under the assumptions that:

1. the s^{th} local model is stable,
 2. the matrix $U \Omega_s$ has full rank,
 3. with an adequate value of i , A^{i-1} is neglected,
- we have:

$$Y_{s,\omega}^i (U \Omega_s)^T (U \Omega_s (U \Omega_s)^T)^{-1} \quad H_{s,i}^d \quad (15)$$

The proof of the theorem can be found in appendix 2.

Now, from equation 15, we have an estimation of the Markov parameter matrix $H_{s,i}^d$.

Remark 1: to estimate the Markov parameter matrix $H_{s,i}^d$ by equation 15, we need the knowledge of the weighting function Ω_s . We use recursive subspace identification to compute this weighting function in the next section.

V. D

It is important to note that the previous result involves the state matrix of the system which depends on the active local model.

The recursive subspace approach is used to determine the switching time between local models. As the local models are stable, the recursive subspace identification can be applied. This identification method does not allow to detect the change in inputs dynamics; therefore if a change

¹The superscript 'd' and 'st' stand for 'deterministic' and 'stochastic' respectively.

is detected it corresponds only to change in the system dynamics. We recommend to the reader, to refer to [4] for the details of the recursive subspace identification procedure.

To begin with, we make a first computation of the weighting function with the recursive subspace algorithm. Then we merge the models which are closed to each other, and we recompute the new weighting functions.

A. The first computation of the weighting functions

The switching time is determined by the recursive subspace identification algorithm. Here we summarize the main stages of the algorithm of Oku and al.[4].

Let us define:

$$\Phi_j = \begin{pmatrix} \varphi_i(1) & \varphi_i(2) & \dots & \varphi_i(j-i) \end{pmatrix} \quad (16)$$

$$\varphi_i(k) = u_i(k) = \begin{matrix} u_k \\ \vdots \\ u_{i+k-1} \end{matrix}, k \in [1, j-i]. \quad (17)$$

$$U_j = \begin{pmatrix} u_i(i) & u_i(i+1) & \dots & u_i(j) \end{pmatrix} \quad (18)$$

Y_j is defined similarly to (18).

Θ_j which is the product of the extended observability matrix and the extended controllability matrix, can be estimated by the formula:

$\Theta_j = Y_j \Pi_{U_j} \Phi_j^T \Psi_j$ with $\Psi_j = \left(\Phi_j \Pi_{U_j} \Phi_j^T \right)^{-1}$ and $\Pi_F = I - F^T (F F^T)^{-1} F$, where I is the identity matrix with the appropriate size and F is a matrix.

If we define P_j as $P_j = (U_j U_j^T)^{-1}$, Θ_j can be obtained also by a usual formulation of the recursive least squares method with a forgetting factor γ ($\gamma < 1$, see [4]):

$$\Theta_j = \Theta_{j-1} - \beta_j (e_j + \Theta_{j-1} q_j) q_j^T \Psi_{j-1}, \quad (19)$$

$$\Psi_j = \frac{1}{\gamma} (\Psi_{j-1} - \beta_j \Psi_{j-1} q_j q_j^T \Psi_{j-1}), \quad (20)$$

$$P_j = \frac{1}{\gamma} (P_{j-1} - \alpha_j P_{j-1} u_i(j) u_i^T(j) P_{j-1}), \quad (21)$$

$$Y_j U_j^T = \gamma Y_{j-1} U_{j-1}^T + y_i(j) u_i(j)^T, \quad (22)$$

$$\Phi_j U_j^T = \gamma \Phi_{j-1} U_{j-1}^T + \varphi_i(j-i) u_i(j)^T, \quad (23)$$

$$\alpha_j = (\gamma + u_i(j)^T P_{j-1} u_i(j))^{-1}, \quad (24)$$

$$\beta_j = \left(\frac{1}{\alpha_j} + q_j^T \Psi_{j-1} q_j \right)^{-1}, \quad (25)$$

$$e_j = y_i(j) - Y_{j-1} U_{j-1}^T P_{j-1} u_i(j)^T, \quad (26)$$

$$q_j = \Phi_{j-1} U_{j-1}^T P_{j-1} u_i(j)^T - \varphi_i(j-i), \quad (27)$$

Let λ be a threshold designed according to the χ^2 -distribution with ℓ_i degrees of freedom. The distance from the estimated parameter Θ_j to the true parameter $\bar{\Theta}_j$, $D(\Theta_j, \bar{\Theta}_j)$ is defined as:

$$D(\Theta_j, \bar{\Theta}_j) =$$

$$\text{Trace} \left(\Theta_j - \bar{\Theta}_j \right) \frac{1}{\sigma^2} \frac{1}{i} \left(\Phi_j \Pi_{U_j} \Phi_j^T \right) \left(\Theta_j - \bar{\Theta}_j \right)^T, \quad (28)$$

where $\hat{Y}_j = Y_j - \hat{Y}_j$, $\hat{Y}_j$ is an estimation of Y_j . An estimation of the variance of the modeling error σ^2 is given by ([4]):

$$\sigma^2_j = \frac{1}{i[(j-i+1) - 2mi]} \times \text{Trace} (Y_j - \Theta_j \Phi_j) \Pi_{U_j} (Y_j - \Theta_j \Phi_j)^T \quad (29)$$

The *change test* is defined as (see [4]):

if $D(\Theta_j, \bar{\Theta}_j) < \lambda$: no change has occurred,

if $D(\Theta_j, \bar{\Theta}_j) > \lambda$: a change has occurred.

Note that in the implementation, the true parameter $\bar{\Theta}_j$ is replaced by an approximation: $\bar{\Theta}_j$ (see [4]) computed as a least squares estimation of $\bar{\Theta}_j$ over a sliding window:

$$\bar{\Theta}_j = \arg \min_{\min} \text{Trace} (\bar{Y} - \bar{\Theta} \bar{\Phi}_j) \Pi_{U_j} (\bar{Y} - \bar{\Theta} \bar{\Phi}_j)^T, \quad (30)$$

with

$$\bar{Y}_j = \begin{pmatrix} y_i(j-L+2i) & \dots & y_i(j) \end{pmatrix} \quad (31)$$

$$\bar{\Phi}_j = \begin{pmatrix} \varphi_i(j-L+i) & \dots & \varphi_i(j-i) \end{pmatrix} \quad (32)$$

Once a change at time instant t is detected, the recursive update equation are re-initialized at time instant t , the re-initialization technique is proposed in [4].

The first computation of the weighting functions is described by the algorithm 1 exposed below.

Algorithm 1 (first computation of the weighting functions):

Step1: (*initial conditions*)

set $s \leftarrow 1$ and $k \leftarrow 0$, where s is the local model index and k is the time index².

Step 2: (*change has not occurred*)

if change doesn't occur (i.e. $D(\Theta_j, \bar{\Theta}_j) < \lambda$) then:

$k \leftarrow k+1$, go to step 4.

Step 3: (*change has occurred*)

if change occurs (i.e. $D(\Theta_j, \bar{\Theta}_j) > \lambda$) then:

$s \leftarrow s+1$ and $k \leftarrow k+1$.

Step 4: (*computation of the weighting function*)

model " s " is active: $\omega_{s,k} \leftarrow 1$ and $\omega_{r,k} \leftarrow 0$, $r = s$.

Step 5: (*test of stop*)

$h \leftarrow s$ (h is the number of identified models)

if $k < q$ then go to step 2,

if $k = q$ then stop.

B. The fusion of model

After the first computation of the weighting function by the preceding algorithm, we estimate the Markov parameter matrix $H_{s,i}^d$ (for the h local models). Then, we seek the models which are 'similar'. Firstly, we estimate the covariance of the Markov parameter matrix $H_{s,i}^d$.

²The left arrow " $\leftarrow$ " denote the replacement of the value of the left hand side, by the right hand side.

B.1 Covariance estimate

The variance of the estimate error σ_y can be unbiasedly estimated by:

$$\sigma_y^2 = \frac{1}{i(j-im)} \text{trace}(\varepsilon \varepsilon^T), \quad (33)$$

where:

$\varepsilon = Y_{s,\omega}^i - Y_{s,\omega}^i$ (see 9), $Y_{s,\omega}^i$ is an estimation of $Y_{s,\omega}^i$.

The variance $\Sigma_{H_{s,i}^d}$ of the output estimated Markov parameter matrix $H_{s,i}^d$ is estimated by the following lemma.

Lemma 1:

Unbiased estimation of $\Sigma_{H_{s,i}^d}$ is given by the formula:

$$\Sigma_{H_{s,i}^d} = i\sigma_y^2 [U\Omega_s(U\Omega_s)^T]^{-1}, \quad (34)$$

Now we can make a test on the models to find the models which are closed to each other.

B.2 The 'similar' models

The aim is to find the models close to each other (or "similar"). For that purpose, we define the distance from the estimated Markov parameters of model s to the parameters of the model r as:

$$d(H_{s,i}^d, H_{r,i}^d) = \text{Trace} \left((H_{s,i}^d - H_{r,i}^d) \Sigma_{H_{s,r}}^{-1} (H_{s,i}^d - H_{r,i}^d)^T \right). \quad (35)$$

with:

$$\Sigma_{H_{s,r}} = \frac{N_s}{N_s + N_r} \Sigma_{H_{s,i}^d} + \frac{N_r}{N_s + N_r} \Sigma_{H_{r,i}^d},$$

where $N_s = \sum_{k=1}^q \omega_{s,k}$ and $N_r = \sum_{k=1}^q \omega_{r,k}$.

If δ is a threshold designed according to the χ^2 -distribution with ℓi degrees of freedom, then we perform the following 'similarity test'

B.21 'similarity test'

The models s and r are similar if:

if $d(H_{s,i}^d, H_{r,i}^d) \geq \delta$: the model s and the model r are not 'similar',

if $d(H_{s,i}^d, H_{r,i}^d) \leq \delta$: the model s and the model r are 'similar'.

B.3 Model merging

If $\mathbb{E}_{s0}$ is the set of models which are "similar" to $s0$, the models belonging to this set are replaced by the new model $s0^*$. The new weighting function for the new model $s0^*$ is: $\omega_{s0,k}^* = \omega_{s,k}$ for $k=1, \dots, q$; moreover we estimate the Markov parameters of the new model $s0^*$.

Remark 2: It is necessary to make a new "similarity test" (see B.21) for the merging of the new model (after a first merging), because it can exist two "similar" local models which have not been detected in the first step, because of

the insufficiency of data used to identify the local models. But, by the new computation of the weighting functions and Markov parameters, the number of measurements used to identify the new local models is greater than those which are used to estimate the previous local models.

The merging of the local models is described by algorithm 2.

Algorithm 2 (models merging):

Step 1: (initialization)

st ← 0;

Step 2: (computation of Markov parameters)

compute the Markov parameters (by theorem 2)

Step 3: (test)

Check the "similarity test" (see B.21), and construct the set of models which are "similar".

Step 4: (replacement of the weighting functions)

For each set of models $\mathbb{E}_s$, if cardinal of $\mathbb{E}_s$ is greater than 1 then:

1) st ← 1;

2) $\omega_{s,k} \leftarrow \omega_{s,k}^*$, and cancel the model r , $r \in \mathbb{E}_s$ and $r = s$.

Step 5: (renumber the local models)

Reorganize local model index.

step 6: (stop test)

If st=1 then go to 1,

If st=0 then stop.

Now we have the final weighting function, we estimate the order and the parameters of each local model. The estimation procedure is described in next section.

VI. E

The goal of this section is to estimate the system order n_s and matrices A_s , B_s , C_s and D_s for each local model.

• The system matrix D_s is directly obtained from the Markov parameter matrix $H_{s,i}^d$ (see 12).

Having the Markov parameters (by theorem 2), one can use the algorithm of Kung [3], Ho and Kalman [1], Era [2], Zeiger and McEwen [6] to estimate the system matrices A_s , B_s and C_s .

In this paper we set $i > 2(n_s + 1)$. Let³ $\nu = \text{integer}(i/2)$.

Following we summarize the Era algorithm (*minimal and balanced realization* [2]).

• Build the Hankel matrices $\mathcal{H}_{\nu,s}^0$ and $\mathcal{H}_{\nu,s}^1$ which contains the Markov parameters and are defined by:

$$\mathcal{H}_{\nu,s}^k = \begin{pmatrix} C_s A_s^k B_s & C_s A_s^{k+1} B_s & \dots & C_s A_s^{k+\nu-1} B_s \\ C_s A_s^{k+1} B_s & C_s A_s^{k+2} B_s & \dots & C_s A_s^{k+\nu} B_s \\ \vdots & \vdots & \ddots & \vdots \\ C_s A_s^{k+\nu-1} B_s & C_s A_s^{k+\nu} B_s & \dots & C_s A_s^{k+2\nu-2} B_s \end{pmatrix} \quad (36)$$

• Make a singular values decomposition of the matrix $\mathcal{H}_{\nu,s}^0$:

³Integer(i) is the integer part of i.

$$\mathcal{H}_{\nu,s}^0 = \begin{pmatrix} U_1 \\ U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & S_2 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} \quad (37)$$

$$U_1 S_1 V_1^T$$

where S_2 contains the neglected singular values.

- The system order is equal to the number of singular values in S_1 .
- The extended observability matrix $\Gamma_{\nu,s} = U_1 S_1^{1/2}$, and the controllability matrix $\mathcal{C}_{\nu,s} = S_1^{1/2} V_1^T$ are computed. Note that:

$$\mathcal{H}_{\nu,s}^0 = \Gamma_{\nu,s} \mathcal{C}_{\nu,s},$$

The matrix $\mathcal{C}_s$ can be determined from the first ℓ rows of $\Gamma_{\nu,s}$.

- The matrix B_s is equal to the first m columns of $\mathcal{C}_{\nu,s}$.
- The matrix A_s is given by the formula:

$$A_s = S_1^{-1/2} U_1^T \mathcal{H}_{\nu,s}^1 V_1 S_1^{-1/2}.$$

VII. T

We consider a third order system described as:

$$x_{k+1} = A_s x_k + B_s u_k$$

$$y_k = C_s x_k + D_s u_k + K e_k$$

where s take the value 1 on the intervals $[1, 999]$ and $[1800, 2499]$ and the value 2 on the intervals $[1000, 1799]$ and $[2500, 3500]$; and we have:

$$A_1 = \begin{pmatrix} 0.32 & 0.31 & 0 \\ -0.32 & 0.31 & 0 \\ 0 & 0 & -0.18 \end{pmatrix}, B_1 = \begin{pmatrix} 0.9 & -0.7 \\ 0.71 & -0.5 \\ 0.8 & 0.47 \end{pmatrix}$$

$$C_1 = \begin{pmatrix} -0.55 & 0.2 & 0.8 \\ 0.45 & 0.3 & 0.58 \end{pmatrix}, D_1 = \begin{pmatrix} 0.97 & 0.63 \\ -0.32 & 0.95 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} -0.1 & -0.4 & 0 \\ 0.5 & -0.4 & 0 \\ 0 & 0 & 0.26 \end{pmatrix}, B_2 = \begin{pmatrix} 0.1 & -0.6 \\ 0.32 & -0.66 \\ 0.3 & 0.82 \end{pmatrix}$$

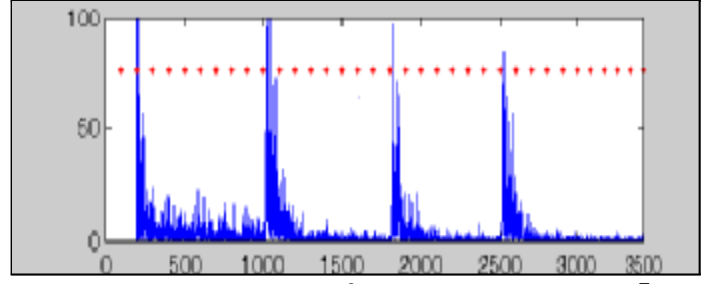
$$C_2 = \begin{pmatrix} -0.8 & -0.1 & 0.7 \\ 0.3 & 0.48 & 0.9 \end{pmatrix}, D_2 = \begin{pmatrix} 0.5 & 0.3 \\ -0.2 & -0.5 \end{pmatrix}.$$

Moreover $u_k \in \mathbb{R}^2$, and $y_k \in \mathbb{R}^2$, the noise $e_k \in \mathbb{R}^2$ is a zero mean white noise and $K = \frac{1}{5} \times I_2$.

We suppose that we have $i+j-1=3500$ measurements of the inputs and outputs. The inputs u_k are white noises.

A. The first computation of the weights (use algorithm 1)

For the implementation of the recursive subspace method, we adopt $i=15$. Since the dimension of the outputs is $\ell=2$, the degree of freedom of the χ^2 distribution is $\ell i=30$. The exponential forgetting factor γ is taken as 0.98 (see [4]).



◆◆: threshold (99.999%, χ^2 distribution); -: $D(\Omega_j, \bar{\Omega}_j)$
Figure 1: the switching times estimated by the recursive subspace algorithm

The recursive subspace algorithm determines the switching times and shows four local models. The first is active in time window $[1:999]$, the second in time window $[1000:1799]$, the third in time window $[1800:2499]$ and the fourth in time window $[2500:3500]$. The switching time is correctly detected.

To compute the parameters for each local model, we set the index i equal to 15 (for each local model identification).

From the parameters obtained with the proposed method, we now estimate the poles for each local model (see figure 2).

100 Monte Carlo realizations are done in each case.

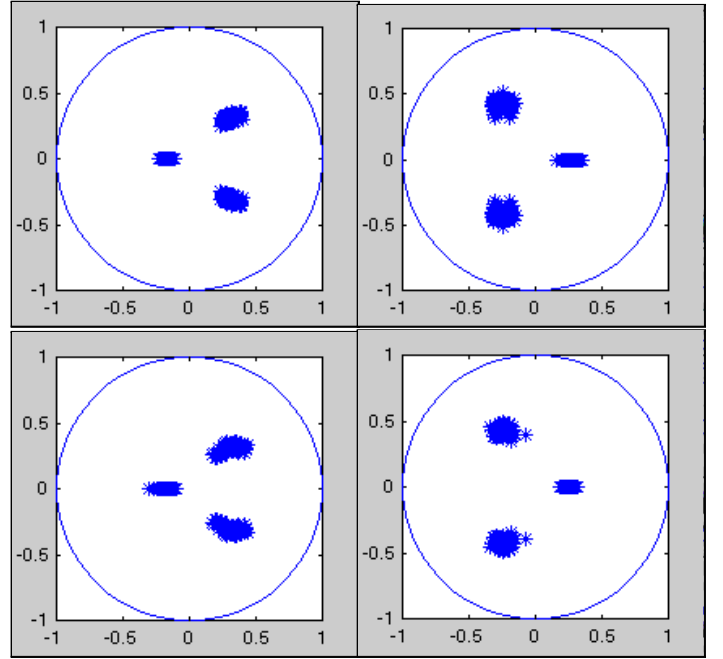


Figure 2: the poles of the four local models according to $\omega_{1,k}$, $\omega_{2,k}$, $\omega_{3,k}$ and $\omega_{4,k}$ respectively.

B. The fusion of the weighting functions (algorithm 2)

The estimated poles show that the local models 1 and 3 are "similar" (figure 2). The same remark hold for local models 2 and 4. That is confirmed by the "similarity test". The distance $d(H_{s,i}^d, H_{r,i}^d)$ of 'similarity test' allows to merge the models without ambiguity. For, hundred experiments we carried out the distance and we have:

$$\blacktriangleright d(H_{1,i}^d, H_{2,i}^d) \in [2300, 2800],$$

- $d(H_{1,i}^d, H_{3,i}^d) \quad [0.5, 4],$
- $d(H_{1,i}^d, H_{4,i}^d) \quad [2000, 2800],$
- $d(H_{2,i}^d, H_{4,i}^d) \quad [1, 4],$

-The threshold (99.999%, χ^2 distribution), $\delta=59$. Then we propose to merge models 1 and 3, that allow to defined a new weight $\omega_{1,k}^* = \omega_{1,k} + \omega_{3,k}$. We also merge the models 2 and 4 and define the new weight: $\omega_{2,k}^* = \omega_{2,k} + \omega_{4,k}$.

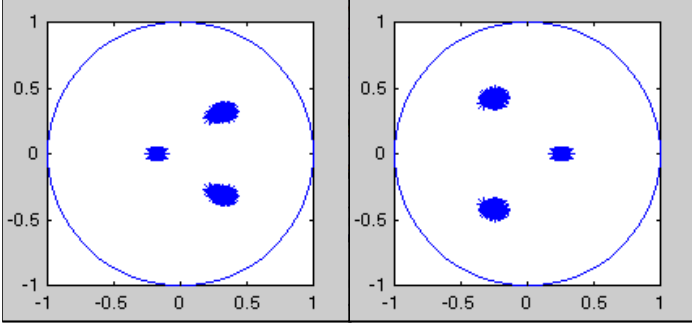


Figure 3: the estimated poles of the 1st and 2nd new local models according to $\omega_{1,k}^*$ and $\omega_{2,k}^*$ respectively.

The figure 3 shows the 100 Monte Carlo realizations for the estimated poles of the new models obtained with $\omega_{1,k}^*$ and $\omega_{2,k}^*$. As we can see, the new weighting functions $\omega_{1,k}^*$ and $\omega_{2,k}^*$ improve the variance of the estimated poles in each case. The estimated poles are unbiased.

VIII. C

Switching model identification is considered in this paper. The technique is based on subspace formulation and uses least squares method for the parameter estimation. The switching model is supposed to be a weighted sum of a local models. A recursive subspace identification technique is used to determine the switching time and a merging algorithm is given to estimate the final weighting function. Finally we estimate the local systems order and parameters from the new estimation of the local models Markov parameters. An illustrating example shows the application of the method.

R

- [1] Ho, B. and Kalman, R. E., *Effective construction of linear state-variate models from input/output functions*. Proceedings. of 3rd Annual Allerton Conf. Circuit and System Theory, pp. 449-459, 1966.
- [2] Juang, C., *Applied system identification*. Prentice -Hall, Englewood Cliffs, NJ, 1994.
- [3] Kung, S. Y., *A new Identification and model reduction algorithm via singular value decompositions*. Proceedings. Twelfth Asilomar Conf. on Circuits, Systems and Computers, pp. 705-714, 1978.
- [4] Oku, H., G. Nijse, M. Verhaegen and V. Verdult. *Change detection in the dynamics with recursive subspace*. Proceedings of the 40th CDC. Orlando, Florida. pp. 2297-2302, 2001.
- [5] Van Overschee, P. and De Moor, B., *Subspace identification for linear systems-Theory, implementation, applications*. Kluwer Academic Publishers, 1996.
- [6] Zeiger, H. P. and McEwen, A. J., *Approximate linear realisations of given dimension via Ho's algorithm*. IEEE Trans. on Auto. Control, pp. 153-155, 1974.

IX. A 1

Proof: The proof of theorem 1 is established in free noise case, the stochastic case is obtained easily.

Form equation 5 we have:

$$\begin{aligned}
 Y_{s,\omega}^i &= \\
 & \left[\begin{array}{cccc} \omega_{s,i} y_{s,i} & \omega_{s,i+1} y_{s,i+1} & \dots & \omega_{s,i+j-1} y_{s,i+j-1} \end{array} \right] \\
 Y_{s,\omega}^i &= \\
 & \left[\begin{array}{cccc} \omega_{s,i} (C_s A_s^{i-1} x_{s,1} + C_s A_s^{i-2} B_s u_1 + \dots + D_s u_i) & \dots & & \\ \omega_{s,i+j-1} (C_s A_s^{i-1} x_{s,j} + C_s A_s^{i-2} B_s u_j + \dots + D_s u_{i+j-1}) & & & \end{array} \right] \\
 Y_{s,\omega}^i &= \\
 & \left[\begin{array}{cccc} \omega_{s,i} C_s A_s^{i-1} x_{s,1} & \dots & \omega_{s,i+j-1} C_s A_s^{i-1} x_{s,j} & \\ \omega_{s,i} (C_s A_s^{i-2} B_s u_1 + \dots + D_s u_i) & \dots & & \\ \omega_{s,i+j-1} (C_s A_s^{i-2} B_s u_j + \dots + D_s u_{i+j-1}) & & & \end{array} \right] + \\
 Y_{s,\omega}^i &= C_s A_s^{i-1} X_s \Omega_s + \\
 H_{s,i}^d & \begin{array}{ccc} \omega_{s,i} u_1 & \dots & \omega_{s,i+j-1} u_j \\ \omega_{s,i} u_2 & \dots & \omega_{s,i+j-1} u_{j+1} \\ \vdots & \vdots & \vdots \\ \omega_{s,i} u_i & \dots & \omega_{s,i+j-1} u_{i+j-1} \end{array}
 \end{aligned}$$

Finally we obtain :

$$Y_{s,\omega}^i = C_s A_s^{i-1} X_s \Omega_s + H_{s,i}^d U \Omega_s$$

X. A 2

Proof:

$$Y_{s,\omega}^i = C_s A_s^{i-1} X_s \Omega_s + H_{s,i}^d U \Omega_s + H_{s,i}^{st} E \Omega_s$$

if A^{i-1} is neglected then:

$$Y_{s,\omega}^i = H_{s,i}^d U \Omega_s + H_{s,i}^{st} E \Omega_s$$

since A_s is assumed to be asymptotically stable, the covariance matrix of the state sequence $(\lim_j \left(\frac{1}{j} (X_s \Omega_s) (X_s \Omega_s)^T \right))$ is bounded. Note that, the elements of the weighting matrix Ω_s take the values 0 and 1.

By the least squares method we can estimate the Markov parameter matrix $H_{s,i}^d$:

$$Y_{s,\omega}^i (U \Omega_s)^T (U \Omega_s (U \Omega_s)^T)^{-1} = H_{s,i}^d$$