15th Triennial World Congress, Barcelona, Spain

STRUCTURE IDENTIFICATION OF MULTIPLE MODELS WITH OUTPUT ERROR LOCAL MODELS

Komi Gasso, Gilles Mourot and José Ragot

Centre de Recherche en Automatique de Nancy - CNRS UMR 7039
CRAN – INPL 2, Avenue de la Forêt de Haye - 54516 Vandoeuvre-lès-Nancy, France
Fax: 03 83 59 56 44 - Email: {ggasso, gmourot, jragot}@ensem.inpl-nancy.fr

Abstract: The paper aims to optimise the structure of a multiple model described by output error local models. Two complementary techniques are proposed for this purpose. The first one deals with the simplification of the local model structure using a pruning approach. The second concerns the optimisation of the number of sub-models and consists in merging neighbouring models that can describe the same behaviour of the system. Their performances are illustrated on simulation examples. *Copyright © 2002 IFAC.*

Keywords: non-linear system, multiple model, structure identification, output error identification, pruning, merging.

1. INTRODUCTION

An approach to modelling non-linear dynamic systems is the multiple model representation (Johansen and Foss, 1993). The basic principle is to represent the system as an interpolation of simple local models. Each sub-model describes the behaviour of the system in a limited part of the operating space. The local validity of a sub-model is specified by an associated weighting function.

The identification of a multiple model involves essentially two tasks: the parameter estimation (local model and weighting function parameters) and the structure identification (characterisation of the operating space, determination of the number of sub-models, identification of their orders and delays).

In most of papers, the identification task is performed using an equation error criterion. The reason is that the criterion is quadratic in the local model parameters which can be computed using the least squares method. But, the identified multiple model may give poor performances when used in simulation. This is a drawback if the model is built for control or diagnosis purposes. To overcome the difficulty, an output error identification is preferable.

The paper deals with the structure optimisation, knowing that the parameter tuning has been already addressed in (Boukhris, *et al.*, 1999; Gasso, *et al.*, 2000). Two complementary techniques are proposed. The first method deals with the structure optimisation of the sub-models. An initial multiple model being identified, the method reduces the complexity of the sub-models by eliminating the irrelevant parameters. The method is an extended study of works partly tackled in (Gasso, *et al.*, 2000). The second method aims to find the appropriate number of local models from a multiple model including a "high" number of sub-models: it consists in merging the neighbouring compatible sub-models which are arbitrary separated by the initial decomposition of the operating space.

The paper is organised as follows: section 2 presents the formulation of the multiple model. Section 3 is devoted to the parameter estimation issues. Section 4 describes the structure optimisation methods which are illustrated on simulation examples in section 5.

2. MATHEMATICAL FORMULATION OF THE MULTIPLE MODEL APPROACH

Consider a non-linear, MISO dynamic system. The multiple model approach represents the system as:

$$y(t) = \sum_{i=1}^{M} w_i\big(\boldsymbol{z}(t), b\big) f_i\big(j\,(t), q_i\big) \qquad \mathbf{(1)}$$

f_i indicates a sub-model. It depends on a regression vector $j(t)$ (t stands for the discrete time) and on a local parameter vector q_i. The weighting function w_i acts as a local validity measure of f_i according to the current operating regime of the system. The weighting functions are defined over the feature (or operating) space **Z** spanned by the vector $\boldsymbol{z}\in\mathbb{R}^{n_z}$. Their number and position determine the partition of **Z**. The vector of feature variables ***z*** can include any variable able to describe the non-linearities of the process. In the paper, the variables ***z*** are *delayed inputs* of the system. The weighting functions are usually bell curve functions (gaussian, sigmoidal functions). They depend on the parameter vector b.

The sub-models could have different functional forms and structures but for the proposed methods, they have necessary the same form. To simplify also the notation, it is assumed that they share the same inputs with the same orders. Thus, the local models are output error models expressed by the relations:

$$y_i(t)=f_i(t)=j^{T}(t)q_i \qquad i=1,\ldots,M \qquad \textbf{(2)}$$

$$j^{T}(t)=[-y(t-1)\,\mathbf{K}\,-y(t-n_y)\;u_1(t-d)\,\mathbf{K}\;u_1(t-d\text{-}n_u)\ldots u_m(t-d)\;\mathbf{K}\;u_m(t-d\text{-}n_u)\;\;1] \qquad \textbf{(3)}$$

The vector $j(t)$ includes the delayed inputs u_r (r=1, …, m) of the system and the delayed output y of the multiple model. n_y and n_u are the orders and d the delay; m is the number of inputs. Using a polynomial notation and according to (3), (2) can be written as:

$$y_i(t)=-A_i(q)y(t)+\sum_{r=1}^{m}B_{ir}(q)u_r(t-d)+q_{i0} \qquad \textbf{(4)}$$

A_i and B_{ir} are polynomials in q^{-1} (delay operator). q_{i0} is a constant (comparable to a bias term).

3. PARAMETER ESTIMATION

Let $q^{T}=[q_1^{T}\;q_2^{T}\;\mathbf{K}\;q_M^{T}]$, the vector of local model parameters. The vectors q and b are estimated by minimising the following output error criterion:

$$J=\sum_{t=h}^{N}\big(y(t)-y_s(t)\big)^2 \quad h=max\big(n_y,n_u+d\big) \qquad \textbf{(5)}$$

y_s is the measured output of the system. The criterion J being non-quadratic in q and b, the optimisation of q and b is realised through the Levenberg-Marquardt (LM) algorithm expressed by the relation:

$$\hat{Q}^{(k+1)}=\hat{Q}^{(k)}-h^{(k)}\mathbf{H}^{-1}\big(\hat{Q}^{(k)}\big)\boldsymbol{G}\big(\hat{Q}^{(k)}\big) \qquad \textbf{(6)}$$

with $Q^{T}=[q^{T}\;b^{T}]$. The approximated hessian **H** and the gradient ***G*** are computed from these equations:

$$\boldsymbol{G}=2\sum_{t=1}^{N}\big(y(t)-y_s(t)\big)\frac{\partial y(t)}{\partial Q}\;,\;\mathbf{H}=2\sum_{t=1}^{N}\frac{\partial y(t)}{\partial Q}\frac{\partial y(t)}{\partial Q^{T}}+l\mathbf{I} \quad \textbf{(7)}$$

They depend on the derivatives of the multiple model output $y(t)$ with respect to q and b. The forms of these sensitivity functions are specified hereafter.

From (1) and (2), the derivative $\partial y(t)/\partial b$ is deduced:

$$\frac{\partial y(t)}{\partial b}=\sum_{i=1}^{M}\left[\frac{\partial w_i\big(\boldsymbol{z}(t),b\big)}{\partial b}y_i(t)+w_i\big(\boldsymbol{z}(t),b\big)\frac{\partial y_i(t)}{\partial b}\right]$$

Using the polynomial form (4), $\partial y_i(t)/\partial b$ is given by:

$$\frac{\partial y_i(t)}{\partial b}=-A_i(q)\frac{\partial y(t)}{\partial b}$$

Combining the two latest equations yields the expression of the sensitive function $\partial y(t)/\partial b$:

$$\big[1+A(q,t)\big]\frac{\partial y(t)}{\partial b}=\sum_{i=1}^{M}\frac{\partial w_i\big(\boldsymbol{z}(t),b\big)}{\partial b}y_i(t) \qquad \textbf{(8-a)}$$

$$A(q,t)=\sum_{i=1}^{M}w_i\big(\boldsymbol{z}(t),b\big)A_i(q) \qquad \textbf{(8-b)}$$

$A(q,t)$ is a time dependant polynomial defined as the weighting sum of the local model polynomials $A_i(q)$.

Similarly, the derivative $\partial y(t)/\partial q$ is derived from (1):

$$\frac{\partial y(t)}{\partial q}=\sum_{i=1}^{M}\left[\frac{\partial w_i\big(\boldsymbol{z}(t),b\big)}{\partial q}y_i(t)+w_i\big(\boldsymbol{z}(t),b\big)\frac{\partial y_i(t)}{\partial q}\right]$$

As the delayed outputs of the multiple model are not considered as feature variables ***z***(t), the derivative $\partial w_i/\partial q$ is null. Therefore, it can be shown using the equations (2), (3) and (4) that $\partial y(t)/\partial q$ becomes:

$$\big[1+A(q,t)\big]\frac{\partial y(t)}{\partial q}=\begin{pmatrix}w_1\big(\boldsymbol{z}(t),b\big)j(t)\\ \mathrm{M}\\ w_M\big(\boldsymbol{z}(t),b\big)j(t)\end{pmatrix} \qquad \textbf{(9)}$$

with $A(q,t)$ defined by (8-b). The relations (8) and (9) provide the dynamic evolution of the sensitivity functions. Notice that the stability of these functions is related to the stability of $[1+A(q,t)]$ i.e. the stability of the multiple model according to (1) and (4).

Using the derivatives, the estimation of q and b is achieved by a two-level algorithm (Boukhris, *et al.*, 1999; Gasso, *et al.*, 2000) because the size of Q is often huge. It consists in computing b by a LM algorithm for q fixed and estimating q by another LM algorithm for b previously determined. The procedure is repeated until convergence. Separating the estimation of b and q simplifies the calculation of the inverse of the hessian matrices (which have consequently a reduced size) and permits to adjust independently the relaxation coefficient h and the regularisation parameter λ for q and b.

4. STRUCTURE IDENTIFICATION

It concerns the choice of the structure of the sub-models, the choice of the feature variables **z** and the partition of the feature space which determines the number of models. In the sequel, the variables **z** are assumed known. The problem treated is twofold: simplification of the structure of the sub-models and optimisation of their number.

4.1. Simplification of the local model structure

In the multiple model representation, simultaneous determination of the structure and the number of local models is difficult because a trade-off must be achieved between the number and the complexity of the local models. Indeed, few complex sub-models are needed to approximate adequately a system and conversely. Therefore, this interdependency forces to proceed sequentially. An elegant way to solve the problem is to fix an initial and common structure for all local models and to identify their appropriate number and position in the operating space (Tanaka, *et al.*, 1995). But after the parameter estimation step, the local models can include superfluous parameters. The reason is that the same structure has been considered for all sub-models whereas it is probable that the regression variables will have different degrees of explanation of the system according to the operating regimes. To fulfil the parsimony principle, the irrelevant parameters can be discarded without altering the generalisation ability of the multiple model. The proposed method is based on a 2nd order development of the criterion *J* around $\hat{\theta}$, that is:

$$J(\theta)\approx J(\hat{\theta})+(\theta-\hat{\theta})^T\boldsymbol{G}(\hat{\theta})+\frac{1}{2}(\theta-\hat{\theta})^T\mathbf{H}(\hat{\theta})(\theta-\hat{\theta}) \quad \textbf{(10)}$$

with $\hat{\theta}$ the estimation of the vector of local model parameters after the parameter tuning. Let $\Delta\theta=(\theta-\hat{\theta})$ and $\Delta J(\Delta\theta)=J(\theta)-J(\hat{\theta})$. Noting that $\boldsymbol{G}(\hat{\theta})=0$, the preceding relation becomes:

$$\Delta J(\Delta\theta)\approx\frac{1}{2}\Delta\theta^T\mathbf{H}(\hat{\theta})\Delta\theta \qquad \textbf{(11)}$$

Simultaneous elimination of *v* parameters of the local models is equivalent to minimise Δ*J* subject to *v* constraints described by the matrix relation:

$$\mathbf{C}_v\left(\hat{\theta}+\Delta\theta\right)=0_{v\times 1},\ \mathbf{C}_v=\begin{bmatrix}0 & 1 & 0 & \ldots & 0\\ & & \vdots & & \\ 0 & \ldots & 1 & 0 & 0\end{bmatrix}\in\mathbb{R}^{v\times n_q}$$

with $n_q = dim(\theta)$. The positions of the elements 1 in the constraint matrix $\mathbf{C}_v$ indicate the parameters to discard. The minimisation of the resultant lagrangian with respect to $\Delta\theta$ provides the solution:

$$\Delta\theta\approx-\mathbf{H}^{-1}\mathbf{C}_v^T\left(\mathbf{C}_v\mathbf{H}^{-1}\mathbf{C}_v^T\right)^{-1}\mathbf{C}_v\hat{\theta}$$

whereof the constrained vector $\hat{\theta}_v^{(c)}$ is deduced:

$$\hat{\theta}_v^{(c)}=\hat{\theta}+\Delta\theta\approx\mathbf{P}_v\hat{\theta} \qquad \textbf{(12-a)}$$

$$\mathbf{P}_v=\mathbf{I}-\mathbf{H}^{-1}\mathbf{C}_v^T\left(\mathbf{C}_v\mathbf{H}^{-1}\mathbf{C}_v^T\right)^{-1}\mathbf{C}_v \qquad \textbf{(12-b)}$$

Notice that to simplify the notation, the hessian $\mathbf{H}(\hat{\theta})$ has been replaced by **H**. The increase of the criterion after the elimination of these *v* parameters is evaluated using the expression:

$$\Delta J_v\approx\frac{1}{2}\hat{\theta}^T\mathbf{C}_v^T\left(\mathbf{C}_v\mathbf{H}^{-1}\mathbf{C}_v^T\right)^{-1}\mathbf{C}_v\hat{\theta} \qquad \textbf{(13)}$$

The deletion of a supplementary parameter (for example the *i*th element of θ) is equivalent to add a new constraint to $\mathbf{C}_v$. This yields:

$$\mathbf{C}_{v+1}=\left[\mathbf{C}_v^T\ \boldsymbol{C}_i^T\right]^T$$

with: $$\boldsymbol{C}_i=[0\ \ldots\ \underbrace{1}_{i^{th}\ position}\ 0\ \ldots\ 0]\in\mathbb{R}^{1\times n_q} \qquad \textbf{(14)}$$

The minimisation of Δ*J* subject to the constraints defined in $\mathbf{C}_{v+1}$ gives the new constrained vector:

$$\hat{\theta}_{v+1}^{(c)}\approx\mathbf{P}_{v+1}\hat{\theta} \qquad \textbf{(15)}$$

$$\mathbf{P}_{v+1}=\left(\mathbf{I}-\mathbf{H}^{-1}\mathbf{P}_v\boldsymbol{C}_i^T\left(\boldsymbol{C}_i\mathbf{P}_v\mathbf{H}^{-1}\boldsymbol{C}_i^T\right)^{-1}\boldsymbol{C}_i\right)\mathbf{P}_v \qquad \textbf{(16)}$$

The new increase of criterion is deduced as:

$$\Delta J_{v+1}\approx\Delta J_v+\frac{1}{2}\hat{\theta}^T\left[\mathbf{P}_v\boldsymbol{C}_i^T\left(\boldsymbol{C}_i\mathbf{P}_v\mathbf{H}^{-1}\boldsymbol{C}_i^T\right)^{-1}\boldsymbol{C}_i\mathbf{P}_v\right]\hat{\theta} \qquad \textbf{(17)}$$

The relations (16) and (17) show that the projection matrix and the variation of criterion after the deletion of *v*+1 parameters can be deduced by recurrence from the deletion of *v* parameters. Hence, an iterative procedure can be applied to discard the irrelevant parameters. But an important feature must be emphasised: there is no guarantee that the parameter variation Δθ calculated after the constrained minimisation conserves the validity of the 2nd order approximation (10) after some iterations. Normally, after the deletion of one parameter, the remaining parameters of the sub-models must be optimised until convergence. Then, the approximation (10) is written again and the deletion of an another parameter is tested. As this basic solution is time consuming, an improvement is proposed as follows: at each iteration *v*, the estimated criterion increase ΔJ_v is compared to the true criterion variation $\Delta J_{real}=J(\hat{\theta}_v^{(c)})-J(\hat{\theta})$ where $J(\hat{\theta}_v^{(c)})$ is computed by simulating the multiple model using the estimation $\hat{\theta}_v^{(c)}$. A large discrepancy between ΔJ_v and ΔJ_{real} means that the 2nd order approximation is no longer valid. Thus, an optimisation of the remaining parameters can be carried out and the elimination procedure is repeated until a stopping criterion is satisfied. This reduces the computational effort. The algorithm for structure refinement of the local models is presented below.

Algorithm 1: parameter elimination

1. Compute the hessian $\mathbf{H}(\hat{\theta})$ from (7) and (9). Set *v*=0, $\mathbf{P}_v=\mathbf{I}_{n_q}$, ΔJ_v=0. Let *S*, the set of non-discarded parameter index.
2. For each parameter $\theta(i)$ $(i\in S)$, generate the constraint vector $\boldsymbol{C}_i$ and compute the variation of criterion ΔJ_{v+1} using eq. (16) and (17).
3. Find the parameter θ(*j*) which deletion produces the minimum increase of criterion noted $\Delta J_{v+1,\,min}$.
4. Compute the constrained parameter vector $\hat{\theta}_v^{(c)}$ using eq. (15) and (16). Compute the true variation of criterion ΔJ_{real} by simulating the resultant multiple model using $\hat{\theta}_v^{(c)}$.

   Test if the stopping criterion based on the true criterion $J(\hat{\theta}_v^{(c)})$ is satisfied. If yes, go to step 6.

5. Test the validity of the 2nd order approximation by comparing DJ_{v+1} and DJ_{real}. If an optimisation is necessary, adjust the remaining parameters of the sub-models by a LM algorithm. Go to step 1.

   Otherwise, calculate the projection matrix $\mathbf{P}_{v+1}$ using eq. (16). Update: $DJ_{v+1} = DJ_{v+1,min}$; $S = S - \{j\}$. Increase v. Go to step 2.
6. Optimise the remaining parameters of the local models by the LM algorithm.

The end of the algorithm is decided on the evolution of a MDL criterion:

$$\text{MDL} = log\left(J(\hat{q}_v^{(c)})\right) + n_{\hat{q}_v^{(c)}} \frac{log(N)}{N} \qquad \textbf{(18)}$$

The degree of freedom of this criterion is the current number of retained parameters i.e. $n_{\hat{q}_v^{(c)}} = n_q - v$.

4.2. Optimisation of the number of local models

The number of local models is related to the partition of the operating space. Different techniques of decomposition can be applied, for example grid partition or k-d partition. But, the drawback of the grid is the curse of dimensionality problem for high dimensional systems. The k-d partition permits to avoid this problem but examples in literature (Nelles, *et al.*, 1999) show that it generates more local models than necessary. Thus, a reduction of the number of sub-models can be realised by merging the models which have been arbitrary separated by the initial partition. The proposed technique is an extension to output error case of a technique developed for serie-parallel models (Gasso, *et al.*, 2001).

Merging two neighbouring sub-models $f_i(\varphi(t),\hat{q}_i)$ and $f_j(\varphi(t),\hat{q}_j)$ is realised by constraining them to have the same behaviour i.e. the same vector of parameters (Gasso, *et al.*, 2001). As the local models have the same structure, the merging consists in determining two vectors Δq_i and Δq_j such as:

$$\hat{q}_i + \Delta q_i = \hat{q}_j + \Delta q_j \qquad \textbf{(19)}$$

This constraint can be written in matrix form:

$$\mathbf{C}_{i,j}\left(\hat{q} + \Delta q\right) = 0 \qquad \textbf{(20-a)}$$

$$\mathbf{C}_{i,j} = [\mathbf{0}_{n\times n} \cdots \underbrace{\mathbf{I}_{n\times n}}_{i^{th}\ position} \mathbf{0}_{n\times n} \cdots \underbrace{-\mathbf{I}_{n\times n}}_{j^{th}\ position} \mathbf{0}_{n\times n} \cdots \mathbf{0}_{n\times n}] \qquad \textbf{(20-b)}$$

$\mathbf{I}_{n\times n}$ and $\mathbf{0}_{n\times n}$ represent respectively the identity matrix and the null matrix of size $n = dim(q_i)$. The parameter variation Δq is calculated by minimising the criterion increase (11) subject to the constraint (20-a). Hence, the merging problem is converted into a least squares optimisation under equality constraint. This is the situation studied in section 4.1. Similarly, the validity of the 2nd order approximation (10) could be lost during the iterations. This must be detected and the parameters of the remaining sub-models optimised.

Algorithm 2: sub-models merging

1. Compute the hessian matrix $\mathbf{H}(\hat{q})$. Generate all candidate pairs (f_i, f_j) of neighbouring sub-models. Let S_{pairs} the set of these pairs. Set v=0, DJ_v=0 and $\mathbf{P}_v = \mathbf{I}_{n_q}$.
2. For each pair of sub-models belonging to S_{pairs},
   - generate the constraint matrix $\mathbf{C}_{i,j}$ (20-b).
   - compute the increase of criterion DJ_{v+1} using eq. (16) and (17) where the matrix $\mathbf{C}_{i,j}$ is substituted to the vector $\boldsymbol{C}_i$.
3. Find the pair (f_a, f_b) of sub-models which merged yields the minimum increase of the criterion. Let $\overline{F}_{a,b}$ the resultant sub-model.
4. Compute the constrained parameter vector $\hat{q}_v^{(c)}$ using eq. (15) and (16) where $\boldsymbol{C}_i$ is replaced by $\mathbf{C}_{i,j}$. Compute the true variation of criterion ΔJ_{real} by simulating the resultant multiple model.

   Test if the stopping criterion based on $J(\hat{q}_v^{(c)})$ is satisfied. If yes, go to step 6.
5. Test the validity of the 2nd order approximation by comparing DJ_{v+1} to ΔJ_{real}. If an optimisation is necessary, adjust the remaining parameters of the local models using a LM algorithm. Go to step 1.

   Otherwise, update the projection matrix $\mathbf{P}_{v+1}$ using eq. (16) where $\mathbf{C}_{i,j}$ is substituted to $\boldsymbol{C}_i$. Update DJ_{v+1}. Increase v.

   Remove (f_a, f_b) from S_{pairs}. Substitute $\overline{F}_{a,b}$ to the sub-models f_a and f_b in any pair where these local models appear. Go to step 2.
6. Optimise the parameters q and b of the reduced multiple model by the LM algorithm.

The stopping condition is based on the evolution of a MDL criterion. The degree of freedom considered is $M_v \times n$ with M_v the current number of sub-models.

The merging of two sub-models f_i and f_j with respectively the weighting functions w_i and w_j yields a sub-model $\overline{F}_{i,j}$ which weighting function is:

$$\overline{w}_{i,j} = w_i + w_j$$

During the merging process, a local model $\overline{F}_{i,j}$ can be combined with a sub-model f_l or with a model $\overline{F}_{a,b}$. To simplify the notation, all the sub-models are noted $\overline{F}_i$. To each $\overline{F}_i$, corresponds a set $\Im_i$ that includes the index of the initial local models. For instance, a sub-model $\overline{F}_i$ related to a set $\Im_i = \{p,q,r\}$ results from the merging of f_p, f_q and f_r. Its weighting function is given by:

$$\overline{w}_i = \sum_{l \in \Im_i} w_l = w_p + w_q + w_r \qquad \textbf{(21)}$$

A singleton set $\Im_i$ indicates that the corresponding local model is not merged with another sub-model.

The merging algorithm requires also the determination of the neighbouring sub-models. The solution to this problem has been developed in (Gasso, *et al.*, 2001) where the reader is referred to.

5. ILLUSTRATIVE EXAMPLES

Two simulation examples are presented to illustrate the proposed methods of structure optimisation.

Example 1: simplification of the structure of the local models

Let the system described by the non-linear equation:

$$y_s(t) = \frac{y_s(t\text{-}1)\left[0.5u_1(t\text{-}1)\text{-}0.3u_2(t\text{-}1)\right]}{1+y_s(t\text{-}1)^2} + 0.3u_1(t\text{-}1)^2 - 0.5u_2(t\text{-}1)^2 + e(t) \qquad \textbf{(22)}$$

where $e(t)$ is an additive noise. The inputs of the systems u_1 and u_2 are formed by the concatenation of piecewise constant signals with variable amplitudes and duration. A set of 1500 data points is used to build the multiple model. The local models are output error types with the initial structure chosen as:

$$y_i(t) = -a_1^{(i)}y(t-1) + b_{11}^{(i)}u_1(t-1) + b_{21}^{(i)}u_2(t-1) + q_{i0}$$

Each sub-model has 4 parameters. The vector of feature variables is $\boldsymbol{z}^T(t) = [u_1(t-1)\ u_2(t-1)]$. The resultant feature space is decomposed in a grid. For this sake, 3 and 2 modalities are defined respectively on the support of z_1 and z_2, leading to 6 sub-models. Notice that these choices correspond to the optimal solution obtained using a trial-error technique. At each modality of a feature variable $z_j(t)$ is affected an individual validity function. The validity functions defined on the support of $z_j(t)$ are built such as they sum to 1. They are given by the formulas:

$$m_{1,j}\left(z_j(t)\right) = 1 - r_{1,j}\left(z_j(t)\right)$$

$$m_{l,j}\left(z_j(t)\right) = r_{l-1,j}\left(z_j(t)\right) - r_{l,j}\left(z_j(t)\right) \qquad l = 2, \mathbf{L}, K_j - 1$$

$$m_{K_j,j}\left(z_j(t)\right) = r_{K_j-1,j}\left(z_j(t)\right) \qquad \textbf{(23-a)}$$

K_j is the number of modalities of z_j and $r_{l,j}$ is a sigmoidal function defined as:

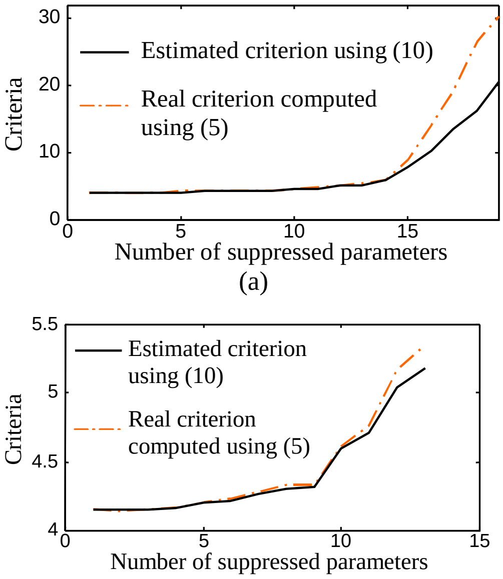
$$r_{l,j}\left(z_j(t)\right) = 1/2 + tanh\left(\frac{z_j(t) - c_{l,j}}{s_{l,j}}\right)\Big/2 \qquad \textbf{(23-b)}$$

$c_{l,j}$ and $s_{l,j}$ are the centre and the dispersion of $r_{l,j}$. A weighting function w_i is expressed by the equation:

$$w_i(\boldsymbol{z}(t), b) = \prod_{j=1}^{n_z} m_{i,j}\left(z_j(t)\right) \qquad \textbf{(24)}$$

with the vector b defined as $b^T = \left[c_{l,j}\ \ s_{l,j}\right]_{l=1,\mathbf{K},K_j}^{j=1,\mathbf{K},n_z}$.

Initially, the validity functions are fixed such as they are evenly spaced. The 24 parameters of the local models are optimised using the LM algorithm. At the convergence, the pruning method is applied. But during the evaluation of the algorithm, the 2nd order validity test is ignored. Fig. 1-a compares the true criterion obtained by simulating the reduced multiple model with the estimated criterion suggested by the 2nd order approximation (10). To make the graphic more legible, a zoom on the first values has been done, giving the graphic of fig. 1-b.



(a)

(b) – Zoom on the first values of the criteria

Fig. 1: Comparison between the true criterion and the estimated one using the 2nd order approximation

It can be noticed that the estimated and the true criteria are very close for the first ten iterations. This confirms the fact that an optimisation of the remaining parameters is not needed at each iteration: it is a time-saving. A significant difference between the criteria occurs from the 12th parameter suggested for deletion. Using the MDL criterion, the algorithm has suggested the elimination of 6 parameters.

Example 2: Reduction of the number of local models

The example illustrates the merging algorithm integrated in a complete procedure of structure optimisation. The procedure can be summarised as:

1. Generate an initial partition of the feature space.
2. Merge the compatible local models using algorithm 2.
3. Optimise the parameters q of the remaining local models $\overline{F}_i$ and the parameters b of the weighting functions in order to improve the approximation abilities of the reduced multiple model.
4. Return to step 2 until satisfaction of a termination criterion.

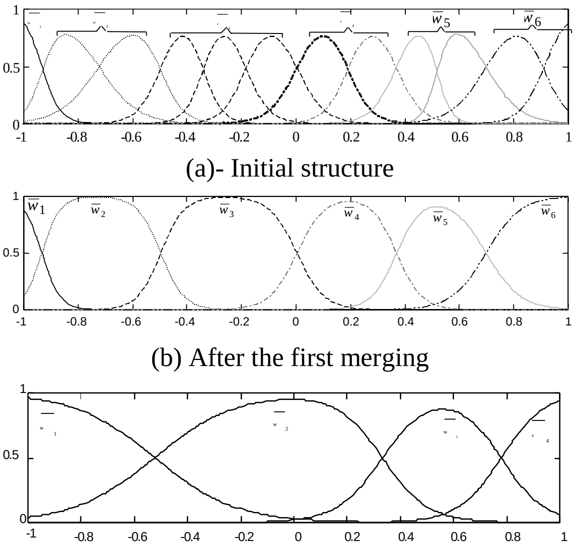
The end of this procedure is decided by analysing the evolution of a cross-validation criterion (using testing data) or a generalisation criterion like the MDL. In this case, the degree of freedom is $dim(b) + n \times M_2$ (total number of parameters involved in the multiple model). M_2 is the current number of sub-models.

To test this complete procedure, consider the system described by the non-linear differential equation:

$$\ddot{y}_s(t) + \dot{y}_s(t) + y_s(t) + y_s(t)^3 = u(t) \qquad \textbf{(25)}$$

The system has been sampled at a period of 0.2s. A white noise has been added to the sampled output y_s. 1500 data points are used for the identification of the multiple model. The feature variable is $\boldsymbol{z}(t)=u(t\text{-}2)$. The regression vector is chosen as

$j^{T}(t) = [-y(t-1)\ -y(t-2)\ u(t-1)\ u(t-2)\ 1]$. The initial multiple model includes 12 sub-models with the corresponding validity functions defined from eq. (23). Notice that as the feature space is mono-dimensional, the validity functions coincide with the weighting functions. Their initial position on the support of $\boldsymbol{z}(t)$ is shown on fig. 2-a.



(a)- Initial structure

(b) After the first merging

(c) Final structure

Fig. 2: Plots of the weighting functions

The parameters of the local models are optimised by the LM algorithm. The merging method is applied next. The neighbours of a sub-model are the models located on its left (if any) and/or its right (if any). Table 1 summarises the results. It shows that after the first iteration, 6 sub-models are obtained. The form and position of the resultant validity functions are plotted on fig. 2-b. The parameters of these functions and those of the resultant sub-models are optimised. The performance of the obtained multiple model is evaluated by computing the MDL criterion. This reduced structure is used as the beginning point for the merging algorithm. By repeating the procedure, the minimum of the MDL is reached for a structure containing 4 sub-models. The validity functions associated to the final structure are shown on fig. 2-c.

Table 1: Results of the structure optimisation method based on the merging algorithm

Iteration	Number of local models	MDL criterion	Identification criterion $(J(\hat{q}_v^{(c)}))$
0	12	1.98	4.90
1	6	1.83	5.15
2	5	1.81	5.24
3	4	1.80	5.35
4	3	1.82	5.67

The identified structure is tested on validation data in order to evaluate its generalisation ability. The mean square error J on these data is 2.05 against 2.47 for the structure with 12 sub-models. The reason is an over-learning phenomenon as the multiple model with 12 sub-models is over-parameterised. The output of the reduced multiple model and the system output are plotted on fig. 3. One can notice the good adequacy of the reduced multiple model.

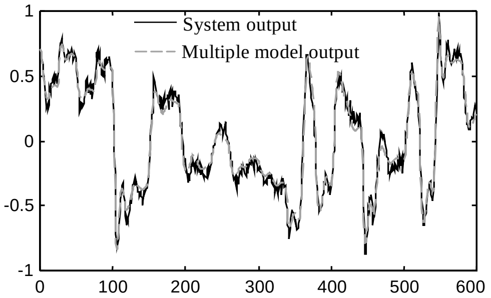


Fig. 3: Performance of the reduced multiple model on the validation data

6. CONCLUSION

In the paper, two techniques are presented for the structure optimisation of a multiple model, considering an output error criterion. The first method concerns the pruning of superfluous parameters of the sub-models whereas the second deals with the optimisation of the number of sub-models using a merging approach. Illustrative examples are provided to show the capacity of the methods. Nevertheless, further investigations must be carried out in order to integrate the two methods in a complete and global procedure of identification.

References

Boukhris, A., G. Mourot and J. Ragot (1999). Non-linear dynamic system identification: a multiple-model approach. *Int. Journal of control*, **vol. 72**, N°7/8, pp. 591-604.

Gasso, K., G. Mourot and J. Ragot (2000). Identification of an output error Takagi-Sugeno model. *Proc. of IEEE Conference on Systems, Man and Cybernetics*, Nashville, USA, 2000.

Gasso, K., G. Mourot and J. Ragot (2001). Structure identification in multiple model representation: elimination and merging of local models. *Proc. of CDC'2001*, Orlando, USA, 2001.

Johansen, T. A. and A. B. Foss (1993). Constructing Narmax using Armax. *Int. Journal of Control*, **vol. 58**, N°5, pp 1125-1153.

Nelles, O., A. Fink, R., Babuška and Setnes M. (1999). Comparison of two construction algorithms for Takagi-Sugeno fuzzy models. *Proc. of Eufit'99*, Aachen, Germany, 1999.

Tanaka, K., M. Sano and H. Watanabe (1995). Modelling and control of carbon monoxide concentration using a neuro-fuzzy technique. *IEEE Trans. on Fuzzy Systems*, **vol. 3**, N°3, pp. 271-279.