A BRANCH-AND-BOUND APPROACH TO THE IDENTIFICATION PROBLEM *

Marc HAEST, Georges BASTIN, Michel GEVERS and Vincent WERTZ
Laboratoire d'Automatique, de Dynamique et d'Analyse des Systèmes
Université Catholique de Louvain
Place du Levant 3
B-1348 Louvain-La-Neuve, Belgium

Abstract - The purpose of the paper is to describe and justify a branch-and-bound approach to system identification that allowed us to reduce significantly the number of model structures investigated by our expert system for process identification.

Key Words - Artificial Intelligence; Branch-and-bound; Expert Systems; System Identification.

1 INTRODUCTION

In this paper, we shall justify and extend a branch-and-bound approach to the identification problem to the case of ARARX model structures. The technique, which has been described by Haest *et al.* (see for example Haest *et al.* 1990b), allows one to explore the model set in a quick and rather systematic way. It enables us to reduce significantly the number of model structures investigated by ESPION, our expert system for process identification (Haest *et al.* 1988a, 1988b, 1990a).

Roughly speaking, the model set is first subdivided into a grid of "equidistant" model structures, where the distance between two model structures is defined as the number of free parameters that must be added or deleted from one to obtain the other. This method, combined with the use of statistical tests, gives valuable information on subsets in which the "best" model structure should be searched for. The other subsets are eliminated. Once a smaller subset that most probably contains the "best" model structure has been identified, it can be mapped or explored in turn at a smaller grid scale and so on until the final grid contains an acceptable solution.

The paper is organized as follows. After having recalled conditions under which model structures are hierarchically nested in Section 2, we show that the model set has a lattice structure in Section 3. This means that, given a structure in the model set, it will always be possible to find another structure in the model set such that a particular dominance relation will hold between them. It also means that if we are given two arbitrary structures in the model set, no relation need to exist between them. The consequences of this property are analyzed in the next four Sections. We show, in Section 4, that all expressions establishing that two model structures are in a particular dominance relation can be automatically replaced by an inequality about their respective prediction error variance. Then, we present the F-test and the encompassing principle in Sections 5 and 6, before describing our branch-and-bound method in Section 7. A simple but detailed example on an ARARX system is developed in Section 8. Finally, the branch-and-bound procedure can be applied, at least theoretically, to the case of more complicated model structures, such as ARMAX or Box-Jenkins. Here, however, we are faced with the typical problems that are associated with the methods used to estimate such structures. Those problems are briefly examined in Section 9, where some other general concluding remarks have also been gathered.

*Most of the results presented in this paper have been obtained within the framework of the "Programme FIRST de Formation et d'Impulsion à la Recherche Scientifique et Technologique du Ministère de la Région Wallonne". The scientific responsibility rests with its authors.

2 WHEN DOES A MODEL STRUCTURE UNDERPARAMETRIZE ANOTHER ?

Throughout the paper, we will use the symbol M to represent the set of all model structures, also called the model set. For two given model structures M_1 and M_2 in this set, we define a dominance relation as follows: the structure M_2 is said to overparametrize the structure M_1, written $M_1 \subseteq M_2$, if the structure M_1 can be obtained from M_2 by forcing part of the latter structure to obey some constraints. If this condition holds, it will equivalently be said that the structure M_1 is an underparametrization of M_2. In other words, for M_1 to be an underparametrization of M_2, we should be able to obtain M_1 as a special case of M_2.

By far the most common way to generate underparame-

trizations of a given model structure is by setting some of its parameters to zero. However, this is equivalent to imposing some constraints on the pole-zero configuration of the original model structure. Setting the last coefficient p_n of a polynomial in the backward shift operator q^{-1}

$$Q(q^{-1}) = p_0 + p_1 q^{-1} + \ldots + p_n q^{-n}$$

to zero is equivalent to pushing away one of its roots at infinity, while setting the first coefficient p_0 to zero is equivalent to forcing one of its roots to be zero. Similarly, forcing an intermediate coefficient $p_i, i \in \{1, \ldots, n-1\}$, to be zero introduces some other constraints on the pole-zero configuration of the original structure.

We can also obtain underparametrizations of a given model structure by imposing some of its poles and zeros to cancel each other. For example, the ARX structure

$$A_2(q^{-1})\, y(t) = B_2(q^{-1})\, u(t) + e(t)$$

overparametrizes all ARARX structures of the form

$$A_1(q^{-1})\, y(t) = B_1(q^{-1})\, u(t) + \frac{e(t)}{D_1(q^{-1})}$$

provided the following inequalities hold

$$d(A_1 D_1) \leq d(A_2)$$
$$d(B_1 D_1) \leq d(B_2)$$

where $d(A_1 D_1)$ stands for the degree of the polynomial $A_1(q^{-1}).D_1(q^{-1})$.

Note, however, that one should be very careful when using similar relations between ARARX structures. For example, the structure

$$S_1 : A_1(q^{-1})\, y(t) = B_1(q^{-1})\, u(t) + \frac{e(t)}{D_1(q^{-1})}$$

with $d(A_1) = 2$, $d(B_1) = 1$, and $d(D_1) = 2$, could be considered at first sight as an underparametrization of

$$S_2 : A_2(q^{-1})\, y(t) = B_2(q^{-1})\, u(t) + \frac{e(t)}{D_2(q^{-1})}$$

with $d(A_2) = 3$, $d(B_2) = 2$, and $d(D_2) = 1$, since it seems enough to impose one pole-zero cancellation in the second structure to obtain the first. Nevertheless, we will not be able to obtain S_1 as a special case of S_2 each time a pair of complex conjugate roots will appear in the noise polynomial of the first structure.

Definition 1: In this paper, we will adopt the convention that a structure M_1 is an underparametrization of another structure M_2, if and only if all models in the structure M_1 can be obtained as special cases of models in the structure M_2. It will also be said that the two model structures are *hierarchically nested.*

3 THE LATTICE STRUCTURE OF THE MODEL SET

It is a trivial matter to show that the set M is a *partial order* under the dominance relation, denoted $(M, \subseteq)$. This simply means that, given a structure M_1 in M, it will always be possible to find another structure M_2 in M such that either $M_1 \subseteq M_2$ or $M_2 \subseteq M_1$ holds, but this also means that if we are given two arbitrary structures M_1 and M_2 in M, no relation need to exist between them. We encountered two such structures in Section 2 (S_1 and S_2).

Moreover, the partial order $(M, \subseteq)$ forms what is called a *distributive lattice.* A distributive lattice is a partial order in which (see for example Gusfield and Irving 1989):

1. Each pair of elements M_1, M_2 has a greatest lower bound denoted by $M_1 \cap M_2$, so that $(M_1 \cap M_2) \subseteq M_1$, $(M_1 \cap M_2) \subseteq M_2$, and there is no element M_0 such that $M_0 \subseteq M_1$, $M_0 \subseteq M_2$ and $(M_1 \cap M_2) \subset M_0$;
2. Each pair of elements M_1, M_2 has a least upper bound denoted by $M_1 \cup M_2$, so that $M_1 \subseteq (M_1 \cup M_2)$, $M_2 \subseteq (M_1 \cup M_2)$, and there is no element M_4 such that $M_1 \subseteq M_4$, $M_2 \subseteq M_4$ and $M_4 \subset (M_1 \cup M_2)$;
3. The distributive laws hold, namely:

$$M_1 \cup (M_2 \cap M_3) = (M_1 \cup M_2) \cap (M_1 \cup M_3)$$
$$M_1 \cap (M_2 \cup M_3) = (M_1 \cap M_2) \cup (M_1 \cap M_3)$$

Obviously, in the context of system identification, one can easily see that the terms "greatest lower bound" and "least upper bound" in the above definitions can be replaced by "greatest common underparametrization" and "least common overparametrization", respectively. It is also clear that this terminology is implicitly linked to the model dimension, which means the number of free parameters in the model structure. Indeed, each parameter in the structure has to be considered as one available degree of freedom and a structure M_1 can never overparametrize another structure M_2 if it has fewer degrees of freedom than M_2.

Every pair of model structures in the model set possess a *greatest common underparametrization.* To obtain the greatest common underparametrization of two given model structures M_1 and M_2, it suffices to construct a new model structure that possesses only parameters that are present *both* in M_1 *and* M_2. For example, the greatest common underparametrization of S_1 and S_2 defined in Section 2, $S_0 = S_1 \cap S_2$, is

$$S_0 : A_0(q^{-1})\, y(t) = B_0(q^{-1})\, u(t) + \frac{e(t)}{D_0(q^{-1})}$$

where $d(A_0) = 2$, $d(B_0) = 1$, and $d(D_0) = 1$. Of course, $M_1 \cap M_2 = M_1$ if $M_1 \subseteq M_2$.

On the other hand, it can happen in certain circumstances that this process gives us a model structure with no parameters at all. For this purpose, we define a fictitious empty model structure $S_\emptyset$ in which all parameters are set to zero. $S_\emptyset$ is the greatest common underparametrization of all model structures in the model set M.

Conversely, every pair of model structures in the model set possess a *least common overparametrization.* To obtain the least common overparametrization of two given model structures M_1 and M_2, it suffices to construct a new model structure that possesses only parameters that are present in M_1 *or* in M_2. For example, the least common overparametrization of S_1 and S_2 defined in Section 2, $S_3 = S_1 \cup S_2$, is

$$S_3 : A_3(q^{-1})\, y(t) = B_3(q^{-1})\, u(t) + \frac{e(t)}{D_3(q^{-1})}$$

where $d(A_3) = 3$, $d(B_3) = 3$ and $d(D_3) = 2$. Again, $M_1 \cup M_2 = M_2$ if $M_1 \subseteq M_2$.

Note that even though the model set is theoretically infinite, it is never really the case on a computer. Indeed, due

to the finite length of the data, there exist some bounds on the delays and the orders of the different model polynomials for our model structures to remain identifiable. Even if we have enough data, the maximal model dimension will remain bounded by the amount of working memory that has been allocated at compilation time. In the sequel of the paper, a model structure obeying the above mentioned restrictions will be called an *admissible* model structure. However, the least common overparametrization of all our admissible model structures in the model set M will probably always be an inadmissible model structure.

Finally, with the rules we have given to build $M_1 \cup M_2$ and $M_1 \cap M_2$, it is not difficult to verify that the *distributive laws* hold effectively. Note also that the concepts developed in this Section can be extended to define the "nearest" model structure in a certain model class of a model structure belonging to another model class. For example, the least ARX overparametrization of the ARARX structure S_1 of Section 2 is

$$A(q^{-1})\,y(t) = B(q^{-1})\,u(t) + e(t)$$

where $d(A) = 4$ and $d(B) = 4$. Conversely, this latter structure is also the greatest ARX underparametrization of all ARMAX structures

$$A(q^{-1})\,y(t) = B(q^{-1})\,u(t) + C(q^{-1})\,e(t)$$

for which $d(A) = 4$, $d(B) = 4$ and $d(C) \geq 0$.

4 WHAT ABOUT THE PREDICTION ERROR VARIANCE ?

The reason why the lattice structure of the model set is so important is that all expressions establishing that two model structures are hierarchically nested can be automatically replaced by an expression about their respective prediction error variances. We have

$$M_1 \subseteq M_2 \Rightarrow \sigma^2(M_1) \geq \sigma^2(M_2)$$

where $\sigma^2(M_i)$ stands for the experimental variance of the prediction errors $\varepsilon_k(M_i)$ resulting from the estimation of M_i on N sampled data:

$$\sigma^2(M_i) = \frac{1}{N}\sum_{k=1}^{N} \varepsilon_k^2(M_i)$$

Unfortunately, the reverse property does not hold. However, we have that

$$\sigma^2(M_1) \geq \sigma^2(M_2) \Rightarrow M_2 \not\subseteq M_1$$

To be convinced that the above implications hold true, one should remember that the prediction error variance is nothing but the output of an optimization procedure during which the parameters of the model are calibrated so as to minimize it. So, if $M_1 \subseteq M_2$, $\sigma^2(M_1)$ is the output of a constrained version of the problem of minimizing $\sigma^2(M_2)$. Thus, we are sure to do at least as well with M_2 as with M_1, since each model in M_1 can be obtained as a special case of models in M_2. This is true independently of the hypotheses that can be made about the data, the only restrictions being that we must use rigorously the same data set to estimate our models, and that we take care not to be misled by local minima.

However, if we choose to compare model structures according to their prediction error variances, we should modify the terminology used to state the lattice structure of the model set as follows:

1. Each pair of elements M_1, M_2 has a least upper bound denoted by $\sigma^2(M_1 \cap M_2)$, so that $\sigma^2(M_1 \cap M_2) \geq \sigma^2(M_1)$, $\sigma^2(M_1 \cap M_2) \geq \sigma^2(M_2)$, and there is no element M_0 such that $M_0 \subseteq M_1$, $M_0 \subseteq M_2$ and $\sigma^2(M_1 \cap M_2) > \sigma^2(M_0)$;
2. Each pair of elements M_1, M_2 has a greatest lower bound denoted by $\sigma^2(M_1 \cup M_2)$, so that $\sigma^2(M_1) \geq \sigma^2(M_1 \cup M_2)$, $\sigma^2(M_2) \geq \sigma^2(M_1 \cup M_2)$, and there is no element M_4 such that $M_1 \subseteq M_4$, $M_2 \subseteq M_4$ and $\sigma^2(M_4) > \sigma^2(M_1 \cup M_2)$.
3. The distributive laws hold, namely:

$$\sigma^2(M_1 \cup (M_2 \cap M_3)) = \sigma^2((M_1 \cup M_2) \cap (M_1 \cup M_3))$$
$$\sigma^2(M_1 \cap (M_2 \cup M_3)) = \sigma^2((M_1 \cap M_2) \cup (M_1 \cap M_3))$$

The prediction error variance of the empty structure $S_\emptyset$ is arbitrarily set to the output signal variance:

$$\sigma^2(S_\emptyset) \triangleq \sigma^2(y(t))$$

This is a natural choice since we are sure to do better with any model structure provided it has at least one free parameter.

As a consequence of the finite length of the data set, we also know that model structures exist somewhere in the model set for which the prediction error variance is exactly zero. However, depending on the number of sampled data, those model structures will generally be inadmissible model structures. This is no problem since those model structures are devoid of any interest.

5 THE F-TEST

We cannot compare models on the sole basis of their prediction error variances. Indeed, since the prediction error variance never ceases to decrease with the number of parameters in the model, this would lead us to choose huge models somewhere on the boundary of the set of admissible model structures.

On the other hand, the following quantity, which is asymptotically F-distributed with $(n_2 - n_1)$ and $(N - n_2)$ degrees of freedom, can be used to check whether the prediction error variance increases significantly when the number of parameters of a given model structure M_2 is decreased from n_2 to n_1, yielding M_1, and N sampled data are available:

$$F = \frac{\sigma^2(M_1) - \sigma^2(M_2)}{\sigma^2(M_2)} \cdot \frac{N - n_2}{n_2 - n_1}$$

Indeed, the following expression holds for n_1 and n_2, where $dim(M_i)$ stands for the number of parameters in the structure M_i:

$$n_i = dim(M_i), i = 1, 2.$$

Here, we become dependent on the hypotheses that can be made about the data and the true system that generated them (see for example Ljung 1987, or Söderström and Stoica 1989).

Definition 2: In the following, a model structure will be considered as an *acceptable* solution if all its underparametrizations yield a significantly worst prediction error variance,

while none of its overparametrizations yield a significantly better variance.

Definition 3: An *optimal* solution, in turn, will be defined as the acceptable solution with the least prediction error variance.

6 THE ENCOMPASSING PRINCIPLE

Finally, the F-test cannot be used to compare non-nested model structures. One may then ask what should be done in the case of the two structures S_1 and S_2 we encountered in Section 2? To cope with this problem, the trick is to use the *encompassing principle* (Mizon and Richard 1986), the key idea of which is to compare both model structures with their least common overparametrization $S_1 \cup S_2$. Once the F-tests have been computed, one for S_1 and $S_1 \cup S_2$, the other for S_2 and $S_1 \cup S_2$, the structure yielding the least value is preferred. A similar procedure could be devised with the greatest common underparametrization $S_1 \cap S_2$.

Now, if one wants to combine the encompassing principle with the use of a particular confidence level, three cases may be encountered in practice. If $S_1 \cup S_2$ is used, we have to consider the following situations:

1. One of the structures, say S_1, is significantly worse than $S_1 \cup S_2$ while the other, S_2 in this case, and $S_1 \cup S_2$ do not differ significantly from each other. S_2, which is said to encompass S_1, should be preferred;
2. Both structures S_1 and S_2 are significantly worse than $S_1 \cup S_2$. In this case we are unable to decide which structure from S_1 or S_2 should be preferred but we are left with a new structure that transcends both S_1 and S_2 in merit;
3. Neither S_1, nor S_2, is significantly worse than $S_1 \cup S_2$. Here, the structure with the least number of parameters should be preferred from a parsimony point of view.

On the other hand, the following cases apply when $S_1 \cap S_2$ is used:

1. One of the structures, say S_1, is significantly better than $S_1 \cap S_2$ while the other, S_2 in this case, and $S_1 \cap S_2$ do not differ significantly from each other. S_1, should be preferred;
2. Both structures S_1 and S_2 are significantly better than $S_1 \cap S_2$. In this case, the structure with the highest value of the F-test should be preferred;
3. Neither S_1, nor S_2, is significantly better than $S_1 \cap S_2$. In this case we are unable to decide which structure from S_1 or S_2 should be preferred but we are left with a new structure that transcends both S_1 and S_2 in merit.

7 BRANCH-AND-BOUND

The lattice structure of the model set provides a convenient way to partition the set of all underparametrizations of a given model structure. Moreover, the prediction error variance of a given model structure represents a lower bound on the best value of the prediction error variance that can be achieved in the set of all its underparametrizations. This allows to tackle the identification problem from a *branch-and-bound* point of view, a problem solving paradigm which has been developed mainly in the context of *integer programming* and *artificial intelligence* (see for example Hillier and Lieberman 1989 or Nemhauser and Wolsey 1988).

One way to implement the technique in the context of system identification is to start in the model set with a root or parent model structure of high dimension. Doing this, we are almost sure to start with an overparametrization of the optimal solution we are looking for. Let us assume that an upper bound σ_u^2 on the prediction error variance of an optimal solution has already been obtained. The set of all the underparametrizations of the starting model structure is first divided into several subsets. This is done, for example, by generating all child model structures that can be obtained by removing a constant number of parameters at a time from each polynomial in the parent model structure. Then, it suffices to estimate these model structures to obtain a lower bound σ_l^2 on the prediction error variance of an hypothetical optimal solution within each of these subsets. Those underparametrizations whose lower bound exceeds the current upper bound and those that are significantly worse than their parent model structure are definitively excluded from further consideration, or pruned, together with *all* their underparametrizations. Then, one of the remaining underparametrizations is chosen to be split further into several subsets. Their lower bounds are obtained in turn and used as before to prune some of them.

An overestimation of the prediction error variance of an optimal solution is obtained the first time a model structure is found all the underparametrizations of which are either significantly worse or already excluded. In the sequel, each time a better acceptable solution is encountered, it replaces the current solution and the upper bound is modified accordingly. Another underparametrization is selected from the remaining ones to be partitioned again, and so on: this process is repeated until an acceptable solution is found, the prediction error variance of which is no greater than all the lower bounds on the remaining subsets. We are sure that this solution is optimal since none of the remaining subsets can contain a better solution.

8 A SIMPLE EXAMPLE ON AN ARARX SYSTEM

Branch-and-bound was run on 1000 sampled data obtained from the following seven parameter, one input - one output ARARX system

$$(1 - 1.2q^{-1} + 0.72q^{-2})\, y(t) = q^{-2}(1 - q^{-1} + 0.5q^{-2})\, u(t) + \frac{e(t)}{1 + q^{-1} + 0.5q^{-2}}$$

where a pseudo random binary signal with amplitude 1 (variance 1) was used for u and a Gaussian white noise with zero mean and variance 0.25 was taken for e. In what follows, the notation $n_a(\tau - n_b)n_d$ has been used to characterize such ARARX structures where n_a and n_d are the number of parameters in the autoregressive and noise polynomials, while τ and n_b are the positions of the first and last nonzero parameters in the exogeneous polynomial. For simplicity, n_d will be omitted in the case of ARX structures.

Beginning from 10 (1-10), an exploration was first conducted in the set of ARX structures. This search is summarized in Table 1 where the stripping factor (s_f), defined as the number of parameters that are removed at a time when attempting to progress from a parent structure, is given for each step (*step*) together with the parent structure number (*from*) and a list of the child structures (*models*) that should be investigated during the step. Only the structures which required estimation have been numbered (n). The prediction error variances (σ^2) of those structures that had to be pruned are followed by a † mark. The prediction error variances of the structures which did *not* require estimation are replaced by a † mark followed by a pair of parentheses enclosing the number of the pruned structure that allowed to avoid the estimation. When the structure has already been estimated, its number is indicated in place of its prediction error variance. Model dimensions are also given in the last column ($dim(\theta)$).

step	s_f	*from*	n	*models*	σ^2	$dim(\theta)$
			1	10(1-10)	0.2560	20
1	5	$n=1$	2	5(1-10)	0.2575	15
			3	10(1-5)	0.2738†	15
			4	10(6-10)	1.2554†	15
2	5	$n=2$	5	0(1-10)	0.3469†	10
				5(1-5)	†($n=3$)	10
				5(6-10)	†($n=4$)	10
3	4	$n=2$	6	1(1-10)	0.3171†	11
			7	5(1-6)	0.2586	11
			8	5(5-10)	1.2634†	11
4	4	$n=7$		1(1-6)	†($n=6$)	7
				5(1-2)	†($n=3$)	7
				5(5-6)	†($n=8$)	7
5	3	$n=7$	9	2(1-6)	0.3286†	8
				5(1-3)	†($n=3$)	8
			10	5(4-6)	1.2644†	8
6	2	$n=7$	11	3(1-6)	0.3024†	9
				5(1-4)	†($n=3$)	9
			12	5(3-6)	1.2644†	9
7	1	$n=7$	13	4(1-6)	0.2586	10
				5(1-5)	†($n=3$)	10
			14	5(2-6)	0.2588	10
8	1	$n=13$		3(1-6)	$n=11$	9
				4(1-5)	†($n=3$)	9
			15	4(2-6)	0.2589	9
9	1	$n=14$		4(2-6)	$n=15$	9
				5(2-5)	†($n=3$)	9
				5(3-6)	$n=12$	9
10	1	$n=15$		3(2-6)	†($n=11$)	8
				4(2-5)	†($n=3$)	8
				4(3-6)	†($n=12$)	8

Table 1: The search in the set of ARX structures.

During the first step, we tried to strip the starting structure of five parameters at a time (*step* = 1, $s_f = 5$, *from* $n = 1$), leading to the estimation of three underparametrizations ($n = 2, 3$ and 4). We could have restarted the procedure from a higher model structure than 10 (1-10) if all the child structures were significantly worse than the root one at this stage. However, since one of these underparametrizations, 5 (1-10) with $\sigma^2 = 0.2575$, does not differ significantly from the root structure ($\sigma^2 = 0.2560$), it is believed here that the latter is an overparametrization of the structure we are looking for. The underparametrizations that differ significantly from the starting one, 10 (1-5) with $\sigma^2 = 0.2738$ and 10 (6-10) with $\sigma^2 = 1.2554$, are pruned forever (†) and the procedure is restarted in step 2 from 5 (1-10), the underparametrization with the best prediction error variance obtained so far (*step* = 2, $s_f = 5$, *from* $n = 2$).

Only one of the three underparametrizations that should be investigated during the second step needs to be estimated since overparametrizations of the others have already been pruned. For example, it makes no sense to compute the prediction error variance associated with 5 (1-5) since this structure is an underparametrization of 10 (1-5), which has been pruned during the first step ($n = 3$). The same holds for 5 (6-10) since this structure can be obtained by removing parameters from 10 (6-10), a structure that has been pruned also during the first step ($n = 4$). At this stage, since the only structure which required estimation during the second step, 0 (1-10) with $\sigma^2 = 0.3469$, must also be pruned, we are unable to go further by stripping five parameters at a time without significantly increasing the prediction error variance.

We then repeat the same procedure during the third step but, now, by trying to strip four parameters at a time (*step* = 3, $s_f = 4$, *from* $n = 2$). Only one promising structure is encountered here: 5 (1-6) with $\sigma^2 = 0.2586$. During the fourth step, we try to strip it of four parameters at a time (*step* = 4, $s_f = 4$, *from* $n = 7$). Without success, all the child structures (1 (1-6), 5 (1-2) and 5 (5-6)) have to be discarded. We then try to strip 5 (1-6) of three parameters at a time in the fifth step (*step* = 5, $s_f = 3$, *from* $n = 7$). Two structures have to be pruned here ($n = 9$ and $n = 10$), while the third one, 5 (1-3), has to be discarded. So, we try to strip 5 (1-6) of two parameters at a time in the sixth step (*step* = 6, $s_f = 2$, *from* $n = 7$). Here again, two structures have to be pruned ($n = 11$ and $n = 12$), while the third one, 5 (1-4), can be ignored since it underparametrizes a pruned structure ($n = 3$). We then enter the seventh step where we try to strip 5 (1-6) of one parameter at a time (*step* = 7, $s_f = 1$, *from* $n = 7$), and so on. The procedure is continued until a dead-end is encountered when trying to strip one parameter at a time (*step* = 10, $s_f = 1$, *from* $n = 15$). After completion of the whole process, the only structure all the underparametrizations of which are significantly worse, while none of its overparametrizations is significantly better, is 4 (2-6) with $\sigma^2 = 0.2589$.

An exploration in the set of ARARX structures can now be started from the structure on which the ARX search stopped. The results are shown in Table 2.

step	s_f	*from*	n	*models*	σ^2	$dim(\theta)$
11	4	$n=15$	16	0(2-2)4	0.3081†	5
12	3	$n=15$	17	1(2-3)3	0.3179†	6
13	2	$n=15$	18	2(2-4)2	0.2592	7
14	2	$n=18$		0(2-2)4	$n=16$	5
15	1	$n=18$		1(2-3)3	$n=17$	6

Table 2: The ARARX search from the best ARX model.

In step 11, we try to remove four degrees of freedom by imposing four poles and zeros of the model structure 4 (2-6) to cancel each other ($s_f = 4$), leading to the estimation of 0 (2-2) 4 with $\sigma^2 = 0.3081$. Since this structure has to be pruned, we then impose three poles and zeros of 4 (2-6) to cancel each other in the next step, and so on. As can be

seen, the ARARX search from the best ARX model ends very quickly on the structure with which the data were generated: 2 (2-4) 2 with $\sigma^2 = 0.2592$. However, this is not very surprising since the ARX search already ended on the least ARX overparametrization of the structure that generated the data. The results are not always as clear-cut. Depending on the number of sampled data and the value of the parameters in the true system, the ARX search can stop on structures that do not overparametrize the true data generating process exactly.

Here, for the sake of clarity, we first restricted the search to the case of ARX structures. One cure would have been to try ARARX structures from the very beginning of the procedure. In this case, 12 ARX and 25 ARARX models were estimated to reach the same conclusion. Since ARARX structures accept least ARX overparametrizations, we then started an exploration in the ARARX set from 10 (1-10) *after* completion of the ARX search and tried to use existing pruned ARX structures to avoid estimating too many ARARX models, but we saved the estimation of only one ARARX model. It could be interesting to first estimate the least ARX overparametrization of any ARARX structure before estimating it, just to see whether the ARARX structure could not already be pruned on a cheaper test.

Finally, we could also have run ARARX explorations only from some of the best ARX structures obtained in lower dimensions. However, one is free to use validation tools here to decide whether or not the exploration can be stopped. In particular, confidence levels on the parameters could be used to check if the walk through the model set could not be pursued further by zeroing some intermediate polynomial coefficients.

9 CONCLUSIONS AND PERSPECTIVES

In the worst case, only 37 ARX and ARARX structures have been estimated. However, a few more could have been investigated if another confidence level had been used for the F-test, but what really matters is that many more could have been generated if another strategy had been used to adapt the number of parameters that are removed at a time. In this case, an exhaustive search over all ARX underparametrizations of the starting structure would have required the estimation of 550 models. This number grows up to 1,864 if all ARARX underparametrizations have to be visited. Since ARARX structures are far more expensive to estimate than their ARX counterparts, one easily sees that considerable savings in time can be gained by pruning the search three judiciously.

Run on industrial data, a strict application of the F-test sometimes leads the branch-and-bound procedure to stop on a structure with too many parameters. This is especially true when the search is limited to the case of ARX model structures. The problem with real-life sampled data is that we are never sure of what can be considered to be the "true generating process" and when it becomes interesting to use more complicated model structures. We have indicated how branch-and-bound can be used to cope with ARARX structures. This strategy, and other scenari that can be imagined in order to save the number of models estimated when working with ARMAX and Box-Jenkins structures, are being studied. Two problems are in order here. First, we have to consider the existence of local minima. An underparametrization M_1 of a given model structure M_2 could yield a better prediction error variance than M_2, simply because the algorithm used to estimate M_2 stopped at a local minimum rather than getting a global one. Therefore, we will have to devise a special mechanism to check such inconsistencies. A simple solution would be to use the parameters obtained for M_1 as initial conditions and to restart an estimation of M_2 from that point. Second, it is well known that overparametrizations can give rise to numerical problems.

One way the branch-and-bound procedure could be brought to stop on more reasonable model structures when run on industrial data, would be to base it on a combination of tools rather than relying only on the values of the F-test. Investigations are currently carried out in order to include classical validation tools in the objective function. For example, the original data set could be separated into two different subsets, one for parameter estimation and the other for model validation. Doing this allows one to compute values of the F-test on the basis of prediction errors obtained on the first data set, while the second data set allows one to test how the estimated model behaves on data that were not used to calibrate its parameters. We also intend to use confidence levels on the parameters to conduct the search more competently and adapt the number of parameters that are removed at a time to map the model set. This could be interesting when working on sampled data from systems with high delays.

REFERENCES

Gusfield, D. and R. W. Irving (1989). *The Stable Marriage Problem. Structure and Algorithms.* The MIT Press, Cambridge, Massachusetts.

Haest, M., G. Bastin, M. Gevers and V. Wertz (1988a). An Expert Workstation for System Identification. *Proc. 8th IFAC/IFORS Symposium on Identification and System Parameter Estimation*, Beijing, P.R.C., Vol. 3, pp. 1990-1995.

Haest, M., G. Bastin, M. Gevers and V. Wertz (1988b). An Expert System for System Identification. *Proc. 1st IFAC Workshop on Artificial Intelligence in Real-time Control*, Swansea, U.K., pp. 101-106.

Haest, M., G. Bastin, M. Gevers and V. Wertz (1990a). Espion: an Expert System for System Identification. *Automatica's Special Issue on Identification and System Parameter Estimation*, Vol. 26, No. 1, pp. 85-95.

Haest, M., G. Bastin, M. Gevers and V. Wertz (1990b). On the use of Search Methodologies in System Identification. *Proc. 29th IEEE Conference on Decision and Control*, Honolulu, Hawaii, Vol. 6, pp. 3182-3187.

Hillier, F. S. and G. J. Lieberman (1989). *Introduction to Operations Research.* McGraw-Hill, New York.

Ljung, L. (1987). *System Identification. Theory for the user.* Prentice Hall, Englewood Cliffs, New Jersey.

Mizon, G. E. and J. F. Richard (1986). The Encompassing Principle and its application to testing non-nested hypotheses. *Econometrica*, Vol. 54, No. 3, pp. 657-678.

Nemhauser, G. L. and L. A. Wolsey (1988). *Integer and Combinatorial Optimization.* Wiley, New York.

Söderström, T. and P. Stoica (1989). *System Identification.* Prentice Hall, Englewood Cliffs, New Jersey.