C 3.3

C 39

AN EXPERT SYSTEM FOR SYSTEM IDENTIFICATION*

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
1348 Louvain-La-Neuve, Belgium

Abstract: An expert system for system identification written with the OPS83 knowledge-based programming language is presented. At the end of an expertise, it provides the user with a set of good models for the system under investigation. If the sampling period used to collect the data seems to be unadapted, the expert system will modify it. An intelligent search through the set of all admissible models is made in order to find the best models of the system. Some validation criteria are used to classify the models and a complete set of facilities is at the user's disposal that allows to modify the expert system behaviour at execution time. One advantage of the expert system approach is that one can not only change decision parameters very easily (such as confidence levels) but one can also change existing rules or add new rules at the price of only one more compilation. Finally, some simulations on data from industrial processes have shown that the expert system behaves just as well as human experts while on simulated noisy data, it finds the true model in the class of ARX or ARARX (also called GLS) models that was used to produce them.

Keywords: Artificial intelligence; computer applications; identification; modelling, parameter estimation; system analysis.

INTRODUCTION

Up to now, system identification has been reserved only to experts in the field. In effect, if many theoretical results are available about estimation methods and their properties, almost no literature is available about methodology. It is left to every individual expert to set up his own strategy as to how to travel through the vast set of candidate model structures in order to arrive at "a best model".

Even the recent emergence of efficient softwares, such as the MATLAB written System Identification Toolbox of Lennart Ljung, has not yet answered the question that remains at the heart of the identification problem: how shall we obtain a good fit to data at low cost, low cost not only meaning the discovery of a model with few parameters, but also the discovery of this model at the price of only moderate computing times and with as little engineering time as possible?

The answer usually is to use as much a priori knowledge about the system as possible, intuition and ingenuity. In addition to the data processing tools, the estimation methods and the validation criteria, the human experts use a lot of heuristic rules acquired through their already long experience in the area and whose application is highly dependent on the characteristics of the data at hand. Those are the reasons it is often heard that identification can hardly be brought into a fully automated procedure.

The situation typically belongs to the ones expert systems claim they are designed for. They are intended to assist or replace man in fields where an insufficiently structured knowledge is admitted for to constitute an accurate, definite and unambiguous working methodology.

In order to assist beginners in the topic, improve the productivity of human experts and give the researchers more insight, an expert system, called ESPION (for Expert System for Process Identification), specifically designed for system identification has been developed at the Department of Automatic Control at Louvain University in Louvain-La-Neuve, Belgium.

* Work sponsored by IRSIA (Belgium).
° Chargé de recherches FNRS (Belgium).

Note that to our knowledge and except for the SEXI Expert System in Identification from the Institut National Polytechnique de Grenoble (S. Gentil and Ph. Conraux, 1985) and the Knowledge Database for System Identification developed by J. E. Larsson and P. Person (1987) at Lund Institute of Technology, little if any had been done on this problem when our project was started.

THE DATA

The expert system can handle multiple input single output systems. At the beginning of an identification session, the user must provide ESPION with a file whose data are stored once and for all in a vector so that no further readings are done in the sequel of the session, resulting in rather substantial savings in time.

Later on, when the inference engine takes over the data handling, the current data set will be divided into two subsets: the first for parameter estimation and the second for model validation. Those data sets will be called, respectively, the estimation data set and the validation data set.

The current data set may differ from the original one depending on possible preliminary data manipulations such as sampling period modifications, data filtering or outliers rejection.

ADMISSIBLE MODELS AND PARAMETER ESTIMATION METHODS

At the present time, the expert system allows the identification of multiple input single output discrete linear ARARX models (for AutoRegressive models with EXogeneous inputs and an AutoRegressive noise model), also called GLS (for Generalized Least Squares) models. They are of the following form

$$A(z^{-1})\, y(t) = \sum_{i=1}^{m} z^{-\tau_i} B_i(z^{-1})\, u_i(t) + \frac{e(t)}{D(z^{-1})} + CC$$

where $A(z^{-1})$, $B_1(z^{-1})$, ... , $B_m(z^{-1})$ and $D(z^{-1})$ are polynomials in the shift operator with $A(z^{-1})$ and $D(z^{-1})$ monic, the τ_i are the dead times, m is the number of inputs, e(t) is a noise driving term and CC holds for an offset term. At any cycle of the inference engine, the user can give the expert system authority to decide whether an offset term should be used or not. Conversely, the user can prohibit the use of this term. The same considerations hold in what concerns the use of a $D(z^{-1})$ polynomial in the denominator of the noise model. Finally, the user can specify to the expert system the signs of some of the static gains known in advance so as to avoid the unnecessary analysis and validation of models estimated with the wrong signs.

THE PRODUCTION SYSTEM FORMALISM

The expert system we have developed with the OPS83 knowledge-based programming language consists of two components: a collection of if-then rules and a global data base called working memory (Ch. Forgy 1986). Each rule contains a conditional expression called the rule's LHS (for Left Hand Side) and an unconditional sequence of actions called the rule's RHS (Right Hand Side). A LHS in turn consists of one or more patterns, a LHS being considered satisfied when every pattern in the LHS matches an element from the working memory.

Recall that the rule interpreter, sometimes called the inference engine, executes a production system by performing a sequence of operations called the recognize-act cycle (Ch. Forgy, 1986). The standard recognize-act cycle is:

1. Match: evaluate the LHS's of the rules to determine which are satisfied given the current contents of the working memory. The rules with satisfied LHS's form a set, called the conflict set.

2. Conflict Resolution: select one rule among the ones with satisfied LHS's. If no rules have satisfied LHS's, halt execution.

3. Act: perform the operations specified in the RHS of the selected rule.

4. Go to step 1.

The changes made during the Act phase of the cycle generally result in a new set of LHS's being satisfied on the next cycle, and this is what gives direction and continuity to the resulting expert system. That is, the typical sequence of events that is seen, as a system runs, is as follows: some rule makes a change to working memory, another rule becomes satisfied as a result of that change, and on the next cycle it is selected and allowed to make further changes. The changes made by the second rule cause a third rule to become satisfied and able to execute.

Processing continues in this manner, each rule responding to the changes made by its predecessors, making changes of its own in an attempt to drive the system closer to the solution it is seeking.

OVERALL BEHAVIOUR OF THE EXPERT SYSTEM

When the user has given the name of the file he wants to work with, when it has been decided to allow the use of an offset term and/or a $D(z^{-1})$ polynomial in the denominator of the noise model and when the signs of the static gains known in advance have been indicated, we enter in the interactive top level of the system. From now on, the user has a complete set of commands that allows him to control the expertise by modifying default parameters such as confidence levels, reading or modifying elements from the working memory or modifying the conflict set. For example, it is possible to execute rules one by one, execute a given number of recognize-act cycles of the inference engine or execute the rules until a particular one will dominate in the conflict set. Nevertheless, it should be noticed that those abilities are especially relevant to the designer and that a normal user will in general ask the expert system to run without his mediation.

The rules of the search "algorithm" are organized around the Fortran tools of the SYSID package (a System Identification library developed in our laboratory), the quotes being justified because it should be remembered that we must depart from the mechanisms of imperative programming as soon as we enter the programming of production systems.

Schematically, the search for the best model is performed at constant sampling period by detecting an elbow in the graph representing the variances of the prediction errors on the estimation data set versus the model dimension, i.e. the number of estimated parameters (L. Ljung 1987). As long as there is no elbow, a fast search is performed for models of increasing dimension.

Before this fast search is applied, an initial structure for the model has to be determined at minimal dimension. As will be seen, this is done together with the obtention of first guesses for the delays of the system.

During the search for the best model structure or thereafter, a quality index is attached to some of the best models, i.e. those that have smallest prediction error variance on the estimation data set. This index is incremented by one each time the model satisfies one particular validation criterion.

As will be seen, when an elbow in the graph of the estimation variance versus the model dimension has been detected, a table is constructed that gives the best models obtained for the current sampling period. In what concerns its adequacy, some rules are used to check if the sampling period can be considered adapted. If this is not the case, the expert system will modify it and verify whether the new sampling period is better than the old one.

TERMINOLOGY

Before the detailed description of the rules, we define here some concepts that will be used frequently in the sequel: admissible model, model dimension, estimation variance and validation variance, identification exercise, exploration, neighbouring structure or model, slow or fast search.

An **admissible model** is a model that possesses a strictly positive integer number of coefficients in each autoregressive and exogenous polynomial and a positive integer, possibly zero, number of coefficients in the noise polynomial.

We call **model dimension** the total number of estimated parameters including possibly the offset term.

For a given model, we call **estimation variance** the experimental variance of the prediction errors computed on the data set used for parameter estimation. We call **validation variance** the experimental variance of the prediction errors computed on the data set used for validation.

An **identification exercise** consists of one entire expertise or model search, including the possible consideration of several different sampling periods.

An **exploration**, in turn, is a model search at constant sampling period. Its aim is to find an optimal model structure for that particular value of the sampling period. So, if the original sampling period is invalidated, an identification exercise will include several explorations. An exploration is performed through the space of all admissible models for structures of increasing complexity until an elbow is detected in the graph between the estimation variance and the model dimension.

A neighbouring structure of a given one is a structure with the same number of unknown parameters as the given one that can be obtained by applying some elementary modifications to it. By **elementary modification**, we mean a modification that alters by only one unity one or two of the integers characterizing the model structure. We can modify the delays, the number of coefficients in the different polynomials and the presence or absence of an offset term. As those modifications must occur at constant dimension, it means that each time the number of coefficients in one particular polynomial is changed, the number of coefficients in another polynomial will have to be changed automatically. Here, the offset term is considered as a coefficient. Such modifications are termed **compensated elementary modifications** or **elementary exchanges**. For example, if we decide to eliminate the offset term from an ARX structure, then we have to add a pole or a zero to the current structure to stay in the same model dimension. Of course, the modifications on the delays do not need to be compensated. We can now define precisely the notion of **neighbouring structure** of a given one as a structure that can be obtained by applying at most one elementary exchange to the given one plus at most one elementary modification on any of the delays at the same time.

At constant model dimension and constant sampling period, we can perform a **fast** or a **slow search**. During such a search, the neighbouring structures of a current one are examined as long as it is possible to find a better one. In the slow search, all the neighbours are examined before retaining the one that led to the best estimation variance to replace the current one from which a new search is restarted. In the fast search, on the other hand, each time a better model is found it replaces immediately the current one from which the search is restarted. When a model is found for which it is not possible to find a better neighbour, the search is stopped for this model dimension and a new search is performed in the next dimension beginning from the model constructed by adding one pole to the best one we obtained in the dimension we just left.

Note that the number of investigated neighbouring structures differs depending on whether we are in a fast search or in a slow one. In the slow search, all the neighbouring structures are considered so that each elementary exchange will be investigated with all the possible modifications that can be applied to the configuration of the delays. In the fast search, we consider the subset of the neighbouring structures where only one elementary modification can be applied at a time, being a compensated or a free one.

For example, when no offset term and no noise polynomial are admitted, the 8 neighbours (black and white spheres) of a very simple single input single output ARX model are shown in Fig. 1 where n and m stand for the number of coefficients in the autoregressive and exogeneous polynomials, respectively, d denotes the delay, a model is characterized by the notation n-m-d, the original model is underlined (grey sphere) and the neighbours considered in a fast search are in bold characters (black spheres).

Needless to say that the slow run will be useful only in special cases such as the obtention of first guesses for the delays at minimal dimension. Indeed, the number of admissible neighbours grows very rapidly when the number of polynomials in the model increases. If we take both an offset term and a noise polynomial into account in the example given above, the maximal number of admissible neighbours becomes 29. So, if we must perform somewhere a search that looks like an exhaustive search, we better apply it for low model dimensions only. In those lower dimensions the maximal number of neighbours is never attained because the borders of the set of all admissible models are reached very easily. The fast search, in turn, is intended for higher model dimensions.

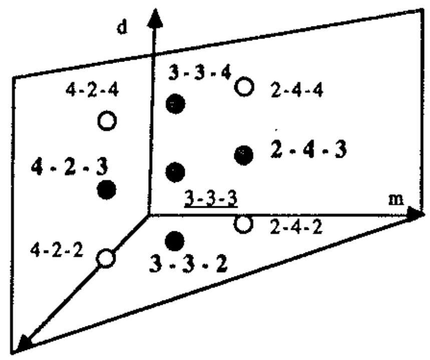

Figure 1: The 8 neighbours of the single input single output ARX model 3-3-3.

THE RULES

We now examine the rules that were implemented. We can distinguish between two categories of rules: the ones that are related to the sailing through the space of all admissible structures towards the "best possible one" and the ones that are related to the analysis of the results, namely the analysis and the validation of the estimated models or the validation of the sampling period.

Some rules make great use of others as is the case for those that obtain initial guesses for the delays by firing several slow searches. One can consider that every set of rules is organized around a particular goal. Some goals that are difficult to obtain can be expressed as an ordered or unordered sequence of simpler ones. So the rules corresponding to complex tasks will have to make several uses of others that correspond to more rudimentary actions following a scenario that will depend on the data at hand.

The fast search rules: Slow searches involve estimating all neighbours of a current model and comparing their estimation variance. The number of neighbouring structures becomes very large as soon as model dimensions are reached for which all neighbours are admissible. As we do not want to consider all possible models of a given dimension, we are constrained to travel through the maze of candidate models in a selective way.

The idea underlying the fast search is as follows: instead of estimating all the neighbouring structures before selecting the one that will replace the current one, they are treated one by one until a structure with a better estimation variance is encountered. Each time this happens, the estimation of the other neighbours is interrupted, the new structure becomes the current one and the computation of its neighbours begins immediately. The fast search is stopped when a structure with no better neighbour is found.

The list of modifications the fast search is allowed to carry on the current structure are characterized by special elements stored into the working memory. They are used to generate the neighbours of the current structure so that one can modify the outline of the study by modifying or destroying those elements corresponding to undesired modifications. This is the way modifications leading to the obtainment of a noise polynomial or an offset term are inhibited when those terms are forbidden by the user. The complete list of modifications that are presently allowed is given hereafter:

- add an offset term by deleting one pole or one zero,
- add one coefficient to the polynomial in the denominator of the noise model by deleting one pole or one zero,
- decrease the delays,
- add one pole by deleting one zero,

- add one zero by deleting one pole or another zero,
- substract one coefficient from the polynomial in the noise denominator by adding one pole or one zero,
- increase the delays,
- delete the offset term by adding one pole or one zero.

Recall that the number of neighbouring structures over which the fast search is done is smaller than for a slow search. But the difference in the number of neighbours that are examined is not the only criterion that distinguishes the fast search from the slow one. The way the neighbours are compared to the current structure is also different. In a fast search, the order in which the neighbours are compared does have an effect upon the overall performances of the expert system: number of estimated models, duration of the identification exercise, discovery of the best structure, ... , etc. In fact, it is here that the whole bulk of the expertise resides, that we approach closest to the fundamental question of the identification problem: how shall we obtain a good fit to data at low cost? At present, the modifications for the fast search are investigated in the order given above. This classification results from tests that have been carried on both industrial and simulated data, but nothing says this ordering is immutable and suitable for all cases. Nevertheless, it is very easy for the user to modify the sequence in which the neighbours are examined in order to compare results and derive conclusions on the optimal strategy to adopt in every case. So we do have at our disposal a true laboratory that allows us to discuss the benefits of one particular scenario with respect to others, and this is where the expert system can acquire real learning abilities.

Recall that the fast search is performed at constant model dimension and constant sampling period. All the estimated models are stored in the working memory so that it is easy to avoid estimating models that have already been computed.

<u>The slow search rules:</u> In the slow search, all the neigbours of the current structure are first estimated. Then, the one that gives the smallest estimation variance is chosen to replace the current one and the cycle is restarted. This time, the order in which the neighbours are examined does not matter, since all the neighbours are computed before starting a new cycle. The search is stopped when a blind alley is encountered, i.e. when a structure with no better neighbours is found. The slow search is also interrupted when a current structure is elected from which a slow search has already started.

Like the fast one, a slow search is performed at constant model dimension and constant sampling period.

<u>The initial guesses for the delays:</u> To prevent the catastrophic increase in the number of admissible models that would be taken into account when the dimension of the current structure grows, we determine the configuration of the delays as soon as possible. To do this, several slow searches are run at minimal dimension from some initial candidate structures.

A model is said to be at minimal dimension if it has exactly one coefficient in each of the autoregressive and exogeneous polynomials, but none in the polynomial of the noise model. So, if we require the presence of at least one parameter in each of the autoregressive and exogeneous polynomials, it is seen that a slow run at minimal dimension can only modify the configuration of the delays. In those conditions, the number of admissible neighbours is considerably reduced.

The candidate structures from which the slow searches are started are obtained on the basis of an analysis of the crosscorrelations between the output and each of the inputs.

At the end of this stage, it is hoped that good approximations for the delays of the system have been obtained or, at least, that some upper bounds on these delays have been established. It should be remembered that this step is an important one. If the initial guess of the delay structure was not done at minimal dimension, it would result in a dramatic increase of the possible cases to be studied in higher dimensions.

Those considerations also hold in what concerns the use of an offset term. The decision whether such a term should be used has implications on the number of admissible models.

<u>The rules that count the number of investigated models:</u> Some rules are used to count at constant sampling period the number of models estimated for one particular model dimension. If the number of computed models is compared to the theoretical maximal number of admissible models at fixed delays, it is possible to characterize the efficiency of the search. Clearly, what we want here is to know how much is gained compared with an exhaustive search.

Note that it is possible to compute the maximal number of admissible models only if it is assumed that the delays are fixed. Otherwise, the number of possible model structures is virtually infinite. At fixed delays, the number of manners we can distribute exactly n parameters in N polynomials is given by

$$\frac{(n + N_{free} - 1)!}{(n - N_{fixed})!\,(N - 1)!}$$

where N_{fixed} holds for the number of polynomials in which the presence of at least one coefficient is required and N_{free} holds for the number of polynomials that may contain no coefficient at all. This expression is only true if no offset term is allowed. If we want to take this term into account, we must add

$$\frac{(n + N_{free} - 2)!}{(n - N_{fixed} - 1)!\,(N - 1)!}$$

admissible models.

In general, the number of estimated models tends to increase dramatically when the delays have not been approximated correctly. There is a high correlation between the number of attempts made by the expert system during the fast search to modify the delays and the number of estimated models. Such a behaviour, where the delays are frequently changed, must be considered a pathological one. When the system seems not to know which way to turn, the sampling period may be suspected to be unadapted, but it may also happen that the system is non linear or non stationary.

<u>The analysis rules:</u> Except for the comparison between the parameters and their standard deviations, the analysis of the estimated models is intended to compute their intrinsic characteristics, which means the characteristics that depend only on the parameters and not on any data: poles, static gains and response times. Later on, those results will be used to validate the sampling period and compute the quality index of the models which led to the best estimation variances.

At constant model dimension, all models with a better estimation variance than the best estimation variance obtained in the preceding dimension are analysed in this sense, i.e. poles, static gains and response times are computed.

<u>The rules that compute the quality index:</u> The goal of model validation is to compute the quality index of the models which led to the best estimation variances.

Normally, a model analysis must precede the model validation. If some models have not yet been analysed, this operation is performed before the validation step.

At constant model dimension, all models with a better estimation variance than the best estimation variance obtained in the preceding dimension are validated, except those having at least one static gain with a wrong sign if prior information on static gains was provided.

The quality index of a model is incremented by one each time the model satisfies one of the following criteria:

- best BIC (see below),
- best validation variance,
- no statistically significant difference between the estimation variance and the validation variance,
- whiteness of the prediction errors in estimation,
- whiteness of the prediction errors in validation,
- independence between the prediction errors in estimation and past inputs,
- independence between the prediction errors in validation and past inputs,
- an elbow has been detected in the graph of the relation between the estimation variance and the model dimension precisely at the dimension of the model being considered,
- good transient behaviour with regard to all the inputs,
- adequacy of the signs of the static gains known a priori,

Here BIC stands for the Bayesian Information Criterion, as proposed by Akaike (H. Akaike, 1978). If σ^2 denotes the estimation variance, $\dim(\theta)$ stands for the model dimension and N is the number of samples, then we have

$$BIC = N \ln(\sigma^2) + \ln(N) \dim(\theta)$$

Some statistical hypothesis tests are used to check that there is no significant difference between the estimation variance and the validation variance or to decide whether the prediction errors are white and independent of past inputs (L. Ljung, 1987), while a model is said to have a bad transient behaviour if one of its time responses exceeds a given treshold expressed as a multiple of the current sampling period. This treshold can be modified by the user at run time.

The sampling period validation rules: If no elbow has been detected in the graph between the estimation variance and the model dimension, the user is warned each time one of the following facts is established by the expert system:

- the exogenous polynomials of all the estimated models are all inconsistent (see below),
- the exogenous polynomials of the best current model are all inconsistent (see below),
- no model with good transient behaviour has been encountered,
- all the estimated models have a dominant pole near the unit circle.

One coefficient is said to be not consistent if the one σ interval around that coefficient contains zero, where σ is the estimated standard deviation of that coefficient. A polynomial is not consistent if all its parameters are inconsistent.

The facts that induce a sampling period modification after the elbow has been detected are given hereafter:

- the exogenous polynomials of the best model at elbow dimension are all inconsistent,
- the best model at elbow dimension has no good transient behaviour,
- the best model at elbow dimension has one pole near the unit circle,
- the system failed to compute all the poles of the best model at elbow dimension.

The classification rules: At the end of an exploration, a table is constructed which gives the best models obtained for different values of the quality index. The principle of parsimony is applied between two models with the same quality index: the one with the least number of parameters is preferred. If more than one model remains with the same quality index and the same number of parameters, the one which gives the smallest prediction error variance is retained.

The exploration rules: An exploration consists of the obtention of initial guesses for the delays at minimal dimension, several fast searches for models of increasing complexity, the detection of an elbow in the graph between the estimation variance and the model dimension and the display of the models with the best quality indices. When a blind alley is encountered in a fast search, a new fast search is started in the next dimension from the model constructed by adding one pole to the best model obtained in the previous dimension. An exploration is made at constant sampling period, but some tests are carried out to check if it can be considered adapted.

The F-test is applied in order to detect the elbow (L. Ljung, 1987). Note that this elbow must be observed for model dimensions differing at least by two to allow the insertion of two complex conjugate poles in the autoregressive polynomial. To avoid problems arising when the estimation variance decreases very slowly for low model dimensions, one does not try to detect an elbow if the model dimension is less than a particular treshold which value can be modified by the user at run time. There is also a maximal model dimension for which the sampling period will be considered unadapted if it is reached without detecting an elbow. Again, this maximal model dimension can be modified by the user at run time.

The identification exercise rules: The rules of this kind organize the whole expertise. They initialize the working memory, assign the default values and create the elements of the working memory that will start the inference engine. They recognize that a sampling period has been judged unadapted and will try another one by starting another exploration on modified data. Those rules also apply some preliminary processing to the data and prepare the estimation and validation data sets. When the expert system decides to end the study, they also install the interface that will enable the user to examine the entire sets of results.

THE CONTROL OF THE EXPERT SYSTEM

What should be kept in mind here is that, even if the control exerted by the exploration and identification rules may seem to be excessive from the production system point of view, the expert system remains basically data-driven. Through the fast search rules, the sampling period validation rules or those that detect an elbow, the behaviour of the expert system is highly sensitive to the characteristics of the data under investigation. If the general profile of the study remains the same from one data file to another, the contents of the working memory, the number of estimated models and their peculiarities, the explored sampling periods and the final model selection will evolve in different ways.

PERFORMANCES

The expert system, run on industrial data from a glass tube drawing bench on which human experts already worked (Wertz and others, 1987), proved to behave as well as them and, in some cases, gave even better results. For example, it took only twenty minutes to identify a model between drawing speed and glass tube diameter in this glass tube production line. 2700 samples were available. We worked with an offset term and, at the end of the fast search in dimension 9, an elbow was detected in dimension 7. 54 models were estimated and 18 of them were validated. The final selection included the structure chosen by the specialists but also some other very good candidates among which a best choice is not clear cut. The final decision remains to the user. Note that in the case of this particular one input-one output system, an exhaustive search would have needed the estimation of 64 models if the delay had been known. An exhaustive search over all models with delays up to the correct one plus one would have required the estimation of 384 models.

The expert system was also run on simulated data obtained from the following 9 parameters, two inputs-one output system

$$(1 + z^{-1} + 0.5\, z^{-2})\, y(t) = z^{-2}\, (1 - z^{-1} + 0.5\, z^{-2})\, u_1(t) + z^{-7}\, (1 + 0.2\, z^{-1})\, u_2(t) + \frac{e(t)}{1 - 0.49\, z^{-2}}$$

where pseudo random binary signals, with amplitudes 1 and 2 respectively, were used for $u_1(t)$ and $u_2(t)$ and a gaussian white noise with zero mean and variance 0.25 was taken for e(t). It took ESPION one hour to perform the expertise on 1000 samples. At the end of the fast search in dimension 11, an elbow was detected in dimension 9 and the correct model structure was found to be the one yielding the highest quality index. 141 models were estimated and 34 of them were validated. The analysis and validation phases are very time consuming relatively to the other phases of the expertise but, here, the increase in computing time is mainly due to the estimation method which becomes an iterative one when ARARX or GLS models are estimated. As expected, the structure with which the data were generated emerged much more clearly from the final selection than for the industrial data. Note that an exhaustive search would have needed the estimation of 495 models assuming the delays were known. Just to give an idea, let us suppose the delays are only known with a precision of one unity and that we want to discuss the effects of all order one modifications on them. Then we would have to estimate about 4455 models. Note also that the model which served to generate the data possesses no less than 116 neighbours.

Table I summarizes the overall behaviour of the expert system for this particular case study. For each investigated model dimension, the starting and final model structures are given, the former in the first position and then the latter. Each model structure is characterized by the number of coefficients in its autoregressive polynomial, n_a, the number of coefficients in the two exogeneous polynomials, n_{b1} and n_{b2}, the two delays of the system, τ_1 and τ_2, the number of coefficients in the denominator polynomial of the noise model, n_d, and the estimation variance σ^2. Finally, the number of estimated models, noted n, is also given for each model dimension. Note that the large number of models investigated at dimension 3 is due to a slow search for the initial guesses of the delays.

In order to mislead the expert system, yet another test has been made on simulated data related to the same system except that the real zeros of the denominator polynomial in the noise model were replaced by a pair of complex conjugate ones

$$(1 + z^{-1} + 0.5\, z^{-2})\, y(t) = z^{-2}\, (1 - z^{-1} + 0.5\, z^{-2})\, u_1(t) + z^{-7}\, (1 + 0.2\, z^{-1})\, u_2(t) + \frac{e(t)}{1 + 0.6\, z^{-1} + 0.18\, z^{-2}}$$

and that a gaussian white noise with zero mean and variance 0.0625 was used for e(t). Despite some problems were expected due both to the particular configuration of the zeros and to the low level of the

dim	n_a	τ_1	n_{b1}	τ_2	n_{b2}	n_d	σ^2	n
3	1	2	1	7	1	0	4.24121	42
	1	3	1	7	1	0	3.60241	
4	2	3	1	7	1	0	1.69535	8
	2	2	1	7	1	0	1.63716	
5	3	2	1	7	1	0	1.63520	12
	2	2	2	7	1	0	0.75233	
6	3	2	2	7	1	0	0.70032	13
	2	2	3	7	1	0	0.44811	
7	3	2	3	7	1	0	0.39520	18
	2	2	3	7	2	0	0.35712	
8	3	2	3	7	2	0	0.35697	8
	2	2	3	7	1	2	0.33691	
9	3	2	3	7	1	2	0.28699	15
	2	2	3	7	2	2	0.26515	
10	3	2	3	7	2	2	0.26506	12
	2	2	3	7	2	3	0.26439	
11	3	2	3	7	2	3	0.26438	13
	3	2	3	7	2	3	0.26438	

Table I: overall behaviour of the expert system

noise, once again, it took about one hour to estimate 139 models and propose the structure which served to generate the data as the best choice.

CONCLUSIONS

The rules and procedures developed in our expert system are quite subjective, but they are the result of many years of experience on both simulated and real-life applications of identification. One major advantage of the expert system approach is that rules and decision parameters can be tested and changed very easily. In future, the classes of admissible structures will be extended to the ones that contain a moving average in the numerator polynomial of the noise model. We also intend to give the user a more user-friendly interface and provide the expert system with real learning capabilities. Finally, the sampling period validation rules have to be strengthened.

REFERENCES

Akaike, H. (1978). On newer statistical approaches to parameter estimation and structure determination. 7-th Triennal World Congress of the IFAC, Helsinki, 1877-1884.

Forgy, Ch. L. (1986). The OPS83 User's Manual - System Version 2.2. Production Systems Technologies, Inc.

Gentil, S., and Ph. Conraux (1985). S.E.XI. Système Expert en Identification. DEA Report, Laboratoire d'Automatique de Grenoble, Institut National Polytechnique de Grenoble.

Larsson, J. E., and P. Persson (1987). A Knowledge Database for System Identification. Department of Automatic Control, Lund Institute of Technology.

Ljung, L. (1987). System Identification: Theory for the User. Prentice-Hall.

Wertz, V., G. Bastin, and M. Haest (1987). Identification of a glass tube drawing bench. Preprints of the 10-th IFAC World Congress on Automatic Control, Vol. 10, 334-339.