VARIABLE SELECTION PROCEDURES AND EFFICIENT SUBOPTIMAL MASK SEARCH ALGORITHMS IN FUZZY INDUCTIVE REASONING

JOSEP MARIA MIRATS I TUR[a,*], FRANÇOIS E. CELLIER[b] and RAFAEL M. HUBER[a]

[a]IRII, Institut de Robòtica i Informàtica Industrial, Universitat Politècnica de Catalunya—Consejo Superior de Investigaciones Científicas, Parc tecnològic de Barcelona-Edifici U, C/Llorens i Artigues, 4–6, 2°, 08028, Barcelona, Spain; [b]Electrical and Computer Engineering Department, The University of Arizona, Tucson, AZ 85721, USA

This paper describes two new suboptimal mask search algorithms for Fuzzy inductive reasoning (FIR), a technique for modelling dynamic systems from observations of their input/output behaviour. Inductive modelling is by its very nature an optimisation problem. Modelling large-scale systems in this fashion involves solving a high-dimensional optimisation problem, a task that invariably carries a high computational cost. Suboptimal search algorithms are therefore important. One of the two proposed algorithms is a new variant of a directed hill-climbing method. The other algorithm is a statistical technique based on spectral coherence functions. The utility of the two techniques is demonstrated by means of an industrial example. A garbage incinerator process is inductively modelled from observations of 20 variable trajectories. Both suboptimal search algorithms lead to similarly good models. Each of the algorithms carries a computational cost that is in the order of a few percent of the cost of solving the complete optimisation problem. Both algorithms can also be used to filter out variables of lesser importance, i.e. they can be used as variable selection tools.

Keywords: Variable selection; Behavioural modelling; Inductive modelling; Fuzzy inductive reasoning; Suboptimal mask search; Hill-climbing

1. INTRODUCTION

Modelling and simulating the output or outputs of a system from its inputs has always been an important task within control engineering. It is of interest to be able to predict, if possible online, the future behaviour of any one or any subset of the system variables (outputs) given their own current and past behaviour as well as the current and past behaviour of any set of additional auxiliary variables (inputs).

Fuzzy inductive reasoning (FIR) (Cellier, 1991) often offers excellent features for dealing with the aforementioned problem. Whereas *deductive* modelling approaches function well in the case of systems whose internal workings are well understood, such as electronic

---

*Corresponding author. Tel.: +34-934015791. Fax: +34-934015750. E-mail: jmirats@iri.upc.es

circuits, *inductive* modelling approaches should be used whenever the internal equations of the system to be modelled are either unavailable, or contain parameters that cannot be accurately estimated. FIR has been successfully employed to modelling such diverse systems as the central nervous control of the human heart (Nebot *et al.*, 1998), the growth patterns of shrimp populations in semi-intensive shrimp farming (Carvajal and Nebot, 1997), the water demand in the City of Barcelona (López, 1999), and the NO_x emission level in a steam boiler (Mirats i Tur *et al.*, 2002).

As explained in Mirats i Tur *et al.* (2002), the modelling engine of FIR determines a so-called *optimal mask*, that indicates, which variables best explain any given output, and how much time delay these variables should have relative to the chosen output. Unfortunately, any algorithm that can find the optimal mask is necessarily of exponential complexity, i.e. the number of masks to be visited during the search for the optimal mask grows exponentially with the number of available input variables, and with the allowed depth of the mask. For this reason, suboptimal search algorithms are necessary for dealing with large-scale systems.

Most ill-defined systems are unfortunately large-scale systems. It is precisely because, in these systems, output variables depend on so many different inputs that these systems are ill-defined. Thus, precisely in those situations where FIR would be most useful, its optimal mask search algorithm cannot be used, because of the sheer size of the systems to be analysed.

It may not be necessary to make use of *all* potential inputs of a system in order to obtain a decent prediction of the output. Different input variables often contain redundant information. Moreover, some of the input information may indirectly be captured in the past history of the output. For example, when predicting the water demand of Barcelona, it may suffice to propose a model that predicts tomorrow's water demand on the basis of today's water demand and the water demand six days ago. This autoregressive model does not list a single input variable directly, yet the dependence on the day of the week is indirectly captured by including the (meanwhile known) water demand six days ago, whereas including today's water demand in the model indirectly captures the dependence on the weather.

Different suboptimal masks may serve almost equally well because of the redundancy contained in the data. Hence, finding the very best of all possible masks may not be a critical need. It may suffice to find a suboptimal mask, as long as it has a quality that is not much lower than that of the optimal mask. Suboptimal mask search strategies have the goal of searching for masks of acceptable quality, while keeping the search space sufficiently small, such that the suboptimal search algorithm is of polynomial rather than exponential complexity.

Suboptimal mask search strategies were studied before. Nebot and Jerez (1997) analyse several variants of hill-climbing algorithms. Hill-climbing algorithms are of polynomial complexity, but often may end up with a suboptimal mask of significantly inferior quality. Jerez and Nebot (1997) analyse the use of genetic algorithms as suboptimal mask search strategies. Unfortunately, genetic algorithms, while sometimes work surprisingly well, cannot be guaranteed to coverage in polynomial time. de Albornoz (1996) discusses a statistical technique based on cross-correlation functions. His algorithm converges in polynomial time, but only looks at linear relationships between variables, and therefore often finds a suboptimal mask of highly inferior quality.

This paper describes two new suboptimal mask search algorithm. The first method is another variant of a hill-climbing technique. It converges more slowly than those algorithms described in Nebot and Jerez (1997), yet it is more likely to come up with a high-quality mask while still converging in polynomial time. The second method is a new variant of a

statistical approach that is based on spectral coherence functions. It also converges in polynomial time, yet contrary to the technique described in de Albornoz (1996), it avoids the pitfall on relying on linear relationships only. Thus, it is more likely to find a high-quality mask.

No suboptimal mask search strategy can be guaranteed to work always and for every example. Therefore, it is important to offer several techniques that can be used in parallel. If all of these techniques find a mask of similar quality, it is likely that these suboptimal masks are good masks with quality values not far different from that of the optimal mask.

2. REDUCING THE MODEL SEARCH SPACE IN FIR

As explained in Mirats i Tur *et al.* (2002), FIR uses a so-called *mask candidate matrix* to define the mask search space. Each " − 1" element within the mask candidate matrix denotes the position of a potential input variable. The optimal mask can be any of the masks consistent with the mask candidate matrix, i.e. that has inputs only where the mask candidate matrix has " − 1" elements. The index used to compare the masks is an entropy-based measure called the *quality of the mask* in the FIR context. The optimality of the mask is evaluated with respect to the maximisation of its predictive power. Details of how the quality of each mask is determined can be found in Cellier (1991).

Let us assume the system has n input variables, and the allowed depth of the mask is d. In this case, a complete mask candidate matrix has $(n \cdot d - 1)$ potential inputs or " − 1" elements. An exhaustive mask search algorithm thus needs to visit $2^{n \cdot d - 1} - 1$ masks and evaluate their qualities. Hence the number of masks to be visited by the algorithm grows exponentially with the number of input variables and with the allowed depth of the mask.

One way to reduce the search space of the mask search algorithm is to reduce the number of " − 1" elements in the mask candidate matrix by eliminating those elements from the list of potential input variables that are less likely to be used in masks of high quality. Both of the suboptimal search strategies described in this paper operate in this fashion. They identify those potential inputs that are more likely to generate high-quality masks, mark those elements in the mask candidate matrix by setting them to " − 1", set all other potential inputs to "0", and then engage the exhaustive search algorithm to systematically evaluate the qualities of the remaining masks.

The first of these techniques operates on the already recoded, i.e. *qualitative*, data. It starts with a mask depth of 1 and performs an exhaustive search. It subsequently increments the depth of the mask, while making use of the results of the previous searches to select, which elements of the mask candidate matrix can be eliminated from the list of potential inputs. Instead of solving one optimal search with a mask candidate matrix of size $d \cdot n$, it solves d separate optimal searches of sizes $n, 2n, \ldots, d \cdot n$, yet with ever more sparsely populated mask candidate matrices. The maximal mask depth d does not need to be pre-selected. The algorithm proposes an optimal value for d.

The second technique operates on the original real-valued, i.e. *quantitative*, data. It proposes a unique sparse mask candidate matrix by computing the delays of each variable that are most closely related with the considered output. This operation is based on the cross-power spectral density of the individual pairs of signals.

2.1. Model Search Space Reduction from Qualitative Data

When using FIR to qualitatively model a system, the maximum *depth* and *complexity* of the mask must be chosen. The maximum mask depth is defined as the number of rows

of the mask candidate matrix.[†] It determines the largest time constant of the system to be captured by the model. However, since only measurement data are available to base the model upon, it may not always be easy to estimate the largest relevant time constant of the system. The proposed suboptimal search algorithm will automatically determine a good value for the mask depth d.

The *complexity*, c, of the mask is the number of non-zero mask elements used in the model.[‡] The mask complexity represents a compromise between the *specificity* and the *predictability* of the model[¶] (Cellier, 1991). A low-complexity mask, i.e. a mask with a small number of m-inputs, makes it easy to make predictions, yet the predictions obtained in this way are not very specific. On the other hand, a complex model, if applicable, can make highly specific predictions; however, there may not be enough evidence gathered from the training data to justify such a prediction. In other words, with growing mask complexity, more and more possible m-input states exist. Since the total number of observations, n_{rec}, remains constant, the observation frequencies of the observed m-input states will become smaller and smaller, until the situation arises that each observed m-input state has been observed exactly once, whereas many legal m-input states have never been observed at all. At this point, the model becomes totally deterministic. Yet, the predictiveness of such a model will be very poor, as already the next predicted state will probably never have been observed before, which invariably makes further predictions impossible. Based on the previous experiences gained with use of the FIR methodology (Cellier *et al.*, 1996; de Albornoz, 1996), a maximum complexity between 5 and 6 offers a good specificity compromise.

The optimal mask search algorithm employed by FIR starts with evaluating the masks of lowest complexity, then proceeds by incrementing the complexity, until the maximum allowed mask complexity, m, has been reached. Even in the exhaustive search algorithm, the maximum complexity is usually specified in order to limit the search space. This compromise is justified, because the algorithm that computes the mask quality punishes high mask complexity sufficiently to make it rather unlikely that a mask of complexity higher than 6 would ever become the optimal mask. Because of this limitation, even the exhaustive search algorithm is strictly speaking of polynomial complexity. The number of masks to be visited is of the order of $(n{\cdot}d - 1)^{m-1}$, i.e. it is polynomial in the number of inputs, n, and the mask depth, d.

The suboptimal search algorithm reduces this number by reducing the number of " − 1" elements in the mask candidate matrix. It starts with a mask of depth $d = 1$. Since at this point in time, no information is known about the system yet, all potential inputs are set to − 1 in the mask candidate matrix. The optimal (exhaustive) mask search algorithm is then employed to evaluate the quality of each mask of complexity $c \leq m$. Masks are grouped in sets of equal complexity. For each complexity, c, the mask of highest quality is found. Its value is Q_{best}. The relative quality of any one of these masks is defined as $Q_{rel} = Q/Q_{best}$. All masks with a relative quality of $Q_{rel} > x$ are considered *good masks*.[§] All good masks of a given complexity are then investigated. If a given input is being used by at least $y\%$ of all good masks of a given complexity, it is considered a *significant input*.[‖]

[†]In some earlier publications, the mask depth was defined as the number of time intervals covered by the mask, which differs from the definition used here by 1.

[‡]In some earlier publications, the complexity of a mask was defined as the number of non-zero mask inputs, sometimes referred to as m-inputs, which differs from the definition used here by 1.

[¶]In FIR, both, the *specificity* and the *predictability* of a model are dependent of the number of data points gathered up from the system as well as of the number of classes that the variables are discretized into.

[§]In the current implementation of the algorithm, $x = 0.975$.

[‖]In the current implementation of the algorithm, $y = 10\%$.

In the next step of the algorithm, all significant inputs of every complexity are marked by " − 1" elements. All insignificant inputs are marked by "0" elements. The depth, d, is now increased by one. Since nothing is known about the significance of inputs at the new row, all elements of that row are marked as " − 1" elements. The exhaustive search is now repeated with the new mask candidate matrix.

For each mask candidate matrix, the optimal mask is determined. Its quality is Q_{opt}. It is the largest quality of all Q_{best} values of all considered complexities. The algorithm continues, until the Q_{opt} value no longer increases when d is incremented.

The algorithm that has been described in this section, like all suboptimal search methods, is a *heuristic* search technique. There is no guarantee that the truly optimal mask will be found in this way. However, the algorithm is based on much experience and a lot of common sense. A significant input indicates that this input is useful in explaining the desired output. It is very likely that the optimal mask makes use of significant inputs only, or at least makes use primarily of significant inputs. Hence it is very likely that the algorithm will either find the optimal mask itself or at least one of insignificantly lower quality.

The proposed termination criterion is also meaningful, but it may sometimes fail. Some systems exhibit cyclic behaviour. For example, the water demand of the city of Barcelona has a strong weekly cycle. Cyclic behaviour can easily be detected by looking at the autocorrelation of each observed variable. If there is a strong cyclic behaviour, the mask depth should be chosen to cover at least one full cycle, i.e. the termination criterion may need to be modified accordingly, by specifying that the algorithm may not terminate until the mask covers at least one full cycle.

2.2. Model Search Space Reduction from Quantitative Data

In this second approach, the original quantitative data rather than its already pre-processed qualitative counterpart is used to accomplish the target of reducing the model search space of FIR.

Each observed variable trajectory can be interpreted as a *signal*, e.g. the fuel flowing through a pipe, contaminated by a superposed *noise*, that is composed of measuring noise, thermal noise, and so on. Hence each of these trajectories can be interpreted as a realisation of a stochastic process. With this interpretation, the energy of the signals can be computed and used to determine, at which delays each input variable contributes a significant amount of energy to the selected output. In this way, the most probable delays of the input variables relative to the output variable can be determined. These can be used to select the positions of " − 1" elements in the mask candidate matrix to be used by the exhaustive search algorithm of FIR.

In order to compute the signal power spectra,[#] it is necessary to work with stationary or at least quasi-stationary stochastic processes. A random signal $x(t_i)$, $t_i \in Z$, is called *strictly stationary* if for every integer I, the $x(t_i + I)$ shifted signal has the same family of distributions, meaning that every distribution remains invariant under a time shift (Kalouptsidis, 1997). For purely random processes, it is necessary as well as sufficient to ensure stationarity that the first-order densities are identical, i.e. that the mean $m(t_i)$ and all moments $m_k(t_i)$ are constant, independent of t_i. This is also necessary but not sufficient for processes with a non-random component, for which, to ensure strict stationarity, shift-invariance of the entire family of densities must be accomplished. As strict stationarity of

---

[#]When referring to the use of the power spectra the author mean the use of the Fourier spectra.

such a process is hard to establish, the concept of stationarity has been relaxed somewhat. A process is called quasi-stationary if its mean value $m(t_i)$ is constant and if the family of second-order densities depends on the time difference $t_i - t_j$ only. Hence a signal $x(t_i)$ is called quasi-stationary if its mean is constant and if its autocorrelation function $R(t_i, t_j)$ depends only on the difference $|t_i - t_j|$.

Signals observed from physical plants always assume values within a limited range. No physical signal ever grows beyond all bounds. Hence, whereas signals may exhibit positive or negative trends temporarily, when averaged over a sufficiently long time span, such signals can always be assumed to be at least quasi-stationary, and thus, spectral analysis can be applied.

The proposed algorithm finds a single sparsely populated mask candidate matrix by computing the delays of each variable that are most relevant to the considered output. This is accomplished by means of a spectral analysis performed on the observed trajectories. To this end, Matlab's *spectrum* function is applied separately to each input/output pair after detrending all variables individually. The spectrum function returns the frequency range, F, over which the power spectra are being sampled. Assuming the time signals were sampled at time intervals of length t_s, the spectra then range from $-f_s/2$ until $+f_s/2$, where $f_s = 1/t_s$. The spectrum lines are equidistantly spaced over the spectral range. The number of spectral lines, n_s, to be chosen is an input parameter to the *spectrum* function. The output parameter F is a vector of length n_s providing the frequency values of the spectral lines.

The system has n observed variables, one of which is the selected output, the remaining $(n - 1)$ variables are potential inputs. The input variables are called $x_1, x_2, \ldots, x_{n-1}$, and the output variable is called y. The *spectrum* function, applied separately to each pair $\langle x_i, y \rangle$, returns, beside from the frequency vector, F, the following power spectra: P_{xxi}, the power spectral density of the input variable x_i; P_{yy}, the power spectral density of the input variable x_i; P_{yy}, the power spectral density of the output variable y; and P_{xyi}, the cross-spectral density of the input/output pair. It also returns the *coherence function*, C_{xyi}, which is defined as follows (Marple, 1987):

$$C_{xyi}(f_i) = \frac{|P_{xyi}(f_i)|^2}{P_{xxi}(f_i)P_{yy}(f_i)}$$

where f_i are the frequency values over which the spectra were sampled. The C_{xyi} values are positive real values in the range $[0, 1]$. They are relative measures of the cross-energy density that exists between the input variable x_i and the output variable y at different frequencies f_i.

After applying the algorithm, $(n - 1)$ coherence functions have been calculated, one for each of the potential inputs. These coherence functions are now used to identify the significant inputs. A *significant input* exists, where its corresponding coherence function exhibits a *significant peak*. The time delays, t_i, associated with these significant peaks are simply the inverses of the frequency values, f_i, at which these peaks occur.

The following *heuristic* algorithm is proposed to identify the significant peaks and significant inputs of the coherence functions:

1. Each coherence function is detrended separately. Since the coherence functions are quasi-stationary, detrending essentially means removing the mean. Matlab's *detrend* function is used to accomplish the task. After detrending the coherence function, the negative values of the detrended coherence function are set to zero, as those values correspond to the smallest of the peaks.

2. If desired, the same process can be repeated several times. Each time, the smallest of the remaining peaks will be cancelled, and only the larger peaks remain.

3. The *significant peaks* are defined as those peaks that are larger than 2.5 times[**] the standard deviation of the detrended coherence function:

$$\text{sig\_peak} = F[\text{find}(C_{xy} > 2.5 * \text{std}(C_{xy}))];$$

4. The *significant inputs* are those delays that correspond to the significant peaks:

$$\text{sig\_inp} = \text{round}(\text{fs} ./ \text{sig\_peak});$$

The significant inputs denote the positions within the mask candidate matrix that need to be filled with " − 1" elements. Due to the limited resolution of the method, the smallest delay obtainable is 2, i.e. no information can be obtained about delays 0 or 1. The algorithm can be completed by either filling the two bottom rows of the mask candidate matrix with " − 1" elements, since the algorithm does not provide any information for that case, or alternatively, by using the first two steps of the previously discussed algorithm to obtain the missing information.

3. QUALITATIVE MODEL OF A GARBAGE INCINERATOR: A MEDIUM-SCALE EXAMPLE

The two previously described algorithms have been applied to an industrial garbage incinerator process, the functioning of which is briefly outlined in Appendix A. In such an incineration system, turbulent combustion with the interaction of fluid mechanics, chemical kinetics, and heat transfer makes the process highly non-linear. For this reason, linear statistical techniques cannot be applied to the task of variable selection. In the example presented here, 19 input variables and one output variable are taken into account. The task to be accomplished is to derive a qualitative model of this process with high predictive power.

The process was sampled in 1-min intervals. Around 43,200 data records were obtained, representing a time period of 30 days. The first 85% of these data records were used to perform the variable selection with the two previously described suboptimal search procedures. The remaining 15% of the data records were used to validate the obtained model.

3.1. Reducing the Mask Search Space Using Qualitative Data

The system under study has 20 variables leading to a large search space of possible masks when an exhaustive search is applied to all possible combination of input variables. Following the procedure for selecting variables outlined in "Model search space reduction from qualitative data", the first mask candidate matrix proposed is of depth $d = 1$, so only static models are being considered. As, at this stage, the number of

---

[**]This criterion could be made more stringent or more relaxed by modifying the value of "2.5". As this value is lowered more and more energy peaks will be found as significant. On the other hand, as this value is increased fewer and fewer energy peaks will be found as significants. A deepest study of how this parameter affects the computation alleviation achieved in the FIR search model task with the given method is planned to be done in a future research.

TABLE I   Static model search results

	Q_{best}	Variables used in masks with $Q > 0.975\,Q_{best}$	Number of masks with $Q > 0.975\,Q_{best}$
C2	0.0214	19	1
C3	0.0643	5,19	1
C4	0.1185	5,6,19	1
C5	0.1773	2,4,5,6,9,19	3
C6	0.2170	1,4,5,6,7,9,12,19	4

masks to be evaluated is still fairly low, the maximum complexity is set to 6. Potential inputs of the proposed mask are set to -1 except for the output element, which is set to $+1$, so that all the possible masks are being considered. The corresponding mask candidate matrix is:

$$(-1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; -1 \;\; +1)$$

The number of -1 elements of the mask is 19. Table I presents for each allowed complexity, the variables that are used as inputs by the *good masks*, i.e. the masks with $Q > 0.975 Q_{best}$.

None of these masks is particularly good. The best masks are those of complexity 6, but even they are of relatively low quality. Evidently, none of the static models will do a very good job predicting the output.

Proceeding to the second step of the algorithm in "Model search space reduction from qualitative data", a mask candidate matrix of depth $d = 2$ is now proposed. Its lower row, i.e. the row corresponding to time t, that is representing inputs without time delay relative to the output, obtains " -1" elements only in those positions, where *significant inputs* were discovered in the previous step of the algorithm, i.e. in columns 1, 2, 4–7, 9, 12 and 19. Since no information is available with respect to the upper row, all of its elements are set to -1. The proposed mask candidate matrix thus takes the form:

$$\begin{pmatrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 0 & -1 & -1 & -1 & -1 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & +1 \end{pmatrix}$$

The new mask candidate matrix has 29 " -1" elements. Since there are now more possible masks, it was decided to reduce the maximum allowed complexity from 6 to 5. Table II shows the inputs used by the *good masks* for each of the allowed mask complexities. Since every mask includes the output variable, the last row shows how many good masks were found for each of the allowed complexities. There were 25 good masks of complexity 3, 313 good masks of complexity 4 and 2009 good masks of complexity 5. The masks of complexity 2 were not presented, because they are trivial. The only good mask of complexity 2 is the one that uses $y(t - \delta t)$ as its one input. The qualities of the good masks of depth $d = 2$ are considerably higher than of the good static masks, because now, the output to be predicted, $y(t)$, may depend on its own past, i.e. the value of the output one sample back, $y(t - \delta t)$, which helps a lot with the prediction.

Every single one of the good masks makes use of $y(t - \delta t)$ as an input. Therefore, the values in the "delay-1" columns of the row entitled "y" are the same as the values in the "delay-0" columns of the same row, which denote the output itself. This search step indicates that most of the 20 variables one time-step back are of similar usefulness for predicting the

TABLE II   Good masks obtained by a depth-2 model search

Delay	Complex 3 ($Q_{best} = 0.5975$) $Q > 0.975\,Q_{best}$ masks		Complex 4 ($Q_{best} = 0.5915$) $Q > 0.975\,Q_{best}$ masks		Complex 5 ($Q_{best} = 0.5927$) $Q > 0.975\,Q_{best}$ masks	
	0	1	0	1	0	1
x_1	1	1	24	24	214	208
x_2	1	1	26	26	232	236
x_3	–	1	–	21	–	200
x_4	1	1	22	22	229	226
x_5	1	1	21	21	195	198
x_6	1	1	26	26	247	241
x_7	1	1	26	26	238	237
x_8	–	1	–	26	–	241
x_9	–	–	11	10	118	123
x_{10}	–	–	–	11	–	98
x_{11}	–	1	–	20	–	184
x_{12}	1	1	25	27	238	239
x_{13}	–	1	–	24	–	225
x_{14}	–	1	–	24	–	208
x_{15}	–	1	–	24	–	225
x_{16}	–	1	–	23	–	224
x_{17}	–	1	–	22	–	219
x_{18}	–	1	–	16	–	187
x_{19}	1	1	26	26	302	295
y	25	25	313	313	2009	2009

output, as they are used almost uniformly by the good masks. There exists a much better discrimination concerning the usefulness of the variables at time t.

For each complexity separately, only those inputs are considered *significant inputs* that are present in at least 10% of the good masks of that complexity. Using this heuristic rule, a mask candidate matrix of depth $d = 3$ can now be constructed. Since no information is available about the usefulness of any of the variables two time-steps back, the top row of the new mask candidate matrix must be filled with " -1" elements.

$$
\begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 0 & -1 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & -1 & -1 \\
-1 & -1 & 0 & -1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & +1
\end{pmatrix}
$$

The new mask candidate matrix contains 41 out of 59 possible " -1" elements. An optimal model search is now performed, proposing this mask candidate matrix to FIR. Table III shows the results of this search. The qualities of the best masks of each complexity are only slightly higher than in the case of the depth-2 models. 4 good masks of complexity 3, 254 good masks of complexity 4, and 4047 good masks of complexity 5 were found. At this point, FIR shows preferences for some variables one time-step back over others. The reason is that FIR now has more choices, and often prefers the same variable two time-steps back.

The way the algorithm is implemented, once an input variable at a certain time delay has been eliminated from one of the mask candidate matrices, it will never show up again in any of the subsequent mask candidate matrices. Therefore, the *discriminator* value (in the example shown here set to 10%) must be chosen carefully, in order not to eliminate

Delay	Complex 3 ($Q_{best} = 0.6113$) $Q > 0.975\,Q_{best}$ masks			Complex 4 ($Q_{best} = 0.5996$) $Q > 0.975\,Q_{best}$ masks			Complex 5 ($Q_{best} = 0.5931$) $Q > 0.975\,Q_{best}$ masks		
	0	1	2	0	1	2	0	1	2
x_1	–	–	–	17	17	17	–	307	322
x_2	1	1	1	24	24	24	405	418	426
x_3	–	–	–	–	–	1	–	–	148
x_4	–	–	–	1	1	1	225	239	250
x_5	–	–	–	–	–	1	–	–	200
x_6	–	–	–	20	20	20	379	415	409
x_7	–	–	–	22	22	22	417	410	408
x_8	–	–	–	–	8	7	–	331	326
x_9	–	–	–	–	–	–	–	–	58
x_{10}	–	–	–	–	–	1	–	–	61
x_{11}	–	–	–	–	–	1	–	–	175
x_{12}	–	–	–	17	20	20	340	410	415
x_{13}	–	–	–	–	20	20	–	406	408
x_{14}	–	–	–	–	4	4	–	237	232
x_{15}	–	–	–	–	14	14	–	304	324
x_{16}	–	–	–	–	1	1	–	257	266
x_{17}	–	–	–	–	1	1	–	203	210
x_{18}	–	–	–	–	–	1	–	–	108
x_{19}	–	–	–	20	20	21	530	530	536
y	4	4	1	254	254	38	4047	4047	154

potentially useful inputs too early. A smaller discriminator might generate a better suboptimal mask at the end, but this goes at the expense of having to evaluate more masks in the process. A higher discriminator value leads to a faster search, but may result in a suboptimal mask of lower quality.

Using the information provided in Table III, a new mask candidate matrix of depth $d = 4$ can now be proposed.

$$
m_{can} = \begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & +1
\end{pmatrix}
$$

In this new candidate mask only 37 " -1 " elements out of 79 possible are used so, leading to a significant reduction of the model search space. This depth-4 candidate mask is given as an input to the modelling engine of FIR, which performs an optimal search in this reduced model search space. Results of this simulation are given in Table IV. Again, a slight increase in the mask qualities is found with respect to the case of the depth-3 models. Models with complexities 2 and 3 are not reported in Table IV. The reason for that is because they are trivial autoregressive models. For the complexity 2 model, the optimal quality was 0.6128 for a mask in which the output depends on itself one time step in the past, i.e. $y = f(y - \delta t)$. In the case of complexity 3, $Q_{best} = 0.6230$, and 2 masks are found that accomplishes the condition $Q > 0.975 Q_{best}$: the first one models the output

TABLE IV   Good masks obtained by a depth-4 model search

Delay	Complex 4 ($Q_{best} = 0.6127$) $Q > 0.975\,Q_{best}$ masks				Complex 5 ($Q_{best} = 0.6088$) $Q > 0.975\,Q_{best}$ masks			
	0	1	2	3	0	1	2	3
x_1	–	–	–	1	–	–	–	30
x_2	2	2	2	2	30	30	30	30
x_3	–	–	–	1	–	–	–	29
x_4	–	–	–	1	–	–	–	27
x_5	–	–	–	1	–	–	–	26
x_6	–	1	1	1	–	31	26	29
x_7	2	2	2	2	29	30	30	30
x_8	–	–	–	1	–	–	–	32
x_9	–	–	–	–	–	–	–	20
x_{10}	–	–	–	1	–	–	–	25
x_{11}	–	–	–	1	–	–	–	26
x_{12}	–	1	1	1	–	29	27	29
x_{13}	–	1	1	1	–	29	30	29
x_{14}	–	–	–	1	–	–	–	31
x_{15}	–	–	–	1	–	–	–	26
x_{16}	–	–	–	1	–	–	–	27
x_{17}	–	–	–	1	–	–	–	28
x_{18}	–	–	–	1	–	–	–	27
x_{19}	1	1	1	1	30	30	30	30
y	41	41	8	33	487	487	–	487

variable as $y(t) = f(y(t-1), y(t-3))$, and the second one uses the input–output model $y(t) = f(y(t-2), y(t-3))$.

The information contained in this Table IV is used to derive a new depth-5 candidate matrix. In this case, the discriminator value (up to now, it has been set to 10%) has to be changed in order not to eliminate potentially useful inputs. Note that if a value of 10% is used for the discriminator parameter no input variables are selected as possible m-inputs to the next model search. Hence, for each complexity separately, only those inputs are considered *significant inputs* that are present in at least 6% of the good masks of that complexity. This discriminator value keeps the variables that seem to be more important in the analysis performed up to this point, yet conserving a reduced search space model. Using this heuristic rule, a mask candidate matrix of depth $d = 5$ can now be constructed.

$$m_{can} = \begin{pmatrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & -1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix}$$

This new proposed candidate mask has 46 " -1 " elements. Results are given in Table V. The best complexity 2 mask is the same that in the previous computation and it will be the same all over the next runs to be performed, so it will not be named again. The models of complexity 3 that satisfies the conditions of the used algorithm explain

TABLE V   Good masks obtained by a depth-5 model search

Delay	Complex 4 ($Q_{best} = 0.6257$) $Q > 0.975\,Q_{best}$ masks					Complex 5 ($Q_{best} = 0.6146$) $Q > 0.975\,Q_{best}$ masks				
	0	1	2	3	4	0	1	2	3	4
x_1	–	–	–	1	1	–	–	–	70	73
x_2	2	2	2	2	2	73	73	73	74	73
x_3	–	–	–	–	–	–	–	–	49	44
x_4	–	–	–	–	–	–	–	–	–	45
x_5	–	–	–	–	–	–	–	–	–	57
x_6	–	1	–	1	1	–	48	–	73	74
x_7	1	1	1	1	1	74	74	74	74	74
x_8	–	–	–	–	–	–	–	–	70	73
x_9	–	–	–	–	–	–	–	–	–	29
x_{10}	–	–	–	–	–	–	–	–	–	35
x_{11}	–	–	–	–	–	–	–	–	–	54
x_{12}	–	1	–	1	1	–	60	–	74	63
x_{13}	–	1	1	1	1	–	73	74	59	75
x_{14}	–	–	–	–	–	–	–	–	73	65
x_{15}	–	–	–	–	–	–	–	–	–	52
x_{16}	–	–	–	–	–	–	–	–	–	58
x_{17}	–	–	–	–	–	–	–	–	–	53
x_{18}	–	–	–	–	–	–	–	–	–	56
x_{19}	1	1	1	1	1	75	–	74	74	74
y	34	34	1	6	29	1228	1228	–	503	729

the output with a maximum found quality of $Q_{best} = 0.6272$ as $y(t) = f(y(t-1), y(t-3))$ and $y(t) = f(y(t-1), y(t-4))$. Again an slight overall increase in the masks quality is obtained.

Notice that, as the procedure is going on, the input variable space is decomposed into the subspaces of most and less related input variables to the output, so allowing a simplification of the model search. Using the information of Table V, again with the discriminator value set to the 6%, the following depth-6 candidate mask is proposed for the next model computation:

$$m_{can} = \begin{pmatrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & -1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix}$$

This mask has again 46 " − 1" elements. Results obtained with this mask are given in Table VI. The obtained complexity-3 models that accomplishes the imposed conditions are simple autoregressive models, with $Q_{best} = 0.6384$. These models are:

$$y(t) = f(y(t-1), y(t-3)); \quad y(t) = f(y(t-1), y(t-4)); \quad y(t) = f(y(t-1), y(t-5))$$

Delay	Complex 4 ($Q_{best} = 0.6301$) $Q > 0.975\,Q_{best}$ masks						Complex 5 ($Q_{best} = 0.6228$) $Q > 0.975\,Q_{best}$ masks					
	0	1	2	3	4	5	0	1	2	3	4	5
x_1	–	–	–	–	1	1	–	–	–	–	60	59
x_2	2	2	2	2	2	2	68	72	69	70	74	73
x_3	–	–	–	–	–	–	–	–	–	–	–	8
x_4	–	–	–	–	–	–	–	–	–	–	–	26
x_5	–	–	–	–	–	–	–	–	–	–	–	20
x_6	–	–	–	1	1	1	–	–	–	43	48	43
x_7	2	2	2	2	2	2	60	60	64	62	62	63
x_8	–	–	–	–	–	–	–	–	–	–	41	64
x_9	–	–	–	–	–	–	–	–	–	–	–	1
x_{10}	–	–	–	–	–	–	–	–	–	–	–	1
x_{11}	–	–	–	–	–	–	–	–	–	–	–	19
x_{12}	–	–	–	1	–	1	–	–	–	64	–	53
x_{13}	–	1	1	–	1	1	–	64	65	–	65	53
x_{14}	–	–	–	–	–	–	–	–	–	44	–	39
x_{15}	–	–	–	–	–	–	–	–	–	–	–	36
x_{16}	–	–	–	–	–	–	–	–	–	–	–	43
x_{17}	–	–	–	–	–	–	–	–	–	–	–	30
x_{18}	–	–	–	–	–	–	–	–	–	–	–	7
x_{19}	1	–	1	1	1	1	60	–	67	63	65	64
y	43	43	–	2	14	30	1080	1080	–	104	419	605

In this new simulation a better discrimination of the variables is obtained. For example, the models satisfying the algorithm conditions have used variables 9 and 10 only once which says that those variables are not important to model the output. The discriminator value used is again the 6% (the lowest integer is used when computing the 6% of the total amount of models accomplishing $Q > 0.975Q_{best}$). Therefore, variables used more than twice, in the case of the complexity 4 models, and more than 64 times, on the complexity 5 models, are selected to propose the next depth-7 candidate mask. This mask has $44 - 1$ elements. Results of the optimal search performed when giving this candidate mask to FIR are given in Table VII.

$$
m_{can} = \begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}
$$

The discriminator value it has been again the 6%. In this simulation more discrimination about variables have been obtained, note that variables x_3, x_4, x_5, x_9, x_{10}, x_{11}, x_{17} and x_{18} only have been used three times each one in a total of 1060 considered models. This suggests that

those variables are not of a big importance in modelling the output of the system. In the light of the obtained results, a new candidate mask of depth-8 is proposed:

$$m_{can} = \begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
-1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}$$

This mask has a total of 47 " -1" elements. Results for this simulation are shown in Table VIII.

The discriminator value it has been again the 6%. As the depth is increased, more and more discrimination about variables is obtained, note that variables x_3, x_4, x_5, x_9, x_{10}, x_{11}, x_{14}, x_{16}, x_{17} and x_{18} only have been used between three and six times each one in a total of 1478 considered models. This suggests that those variables are not of a big importance in modelling the output of the system. In the light of the obtained results, a new candidate mask of depth-9 is proposed:

$$m_{can} = \begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
-1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}$$

In this new candidate mask there are 48 elements set to -1 out of 179. The achieved simplification is considerable; results are shown in Table IX.

The adopted criterion for the discriminator parameter of the algorithm is the same as in the previous simulations. From Table IX a new depth-10 candidate mask is proposed to the FIR

TABLE VII   Good masks obtained by a depth-7 model search

Delay	Complex 4 (Q_{best} = 0.6319) Q > 0.975 Q_{best} masks							Complex 5 (Q_{best} = 0.6250) Q > 0.975 Q_{best} masks						
	0	1	2	3	4	5	6	0	1	2	3	4	5	6
x_1	–	–	–	–	–	–	–	–	–	–	–	–	–	75
x_2	3	3	3	3	3	3	3	72	73	73	71	77	76	73
x_3	–	–	–	–	–	–	–	–	–	–	–	–	–	3
x_4	–	–	–	–	–	–	–	–	–	–	–	–	–	3
x_5	–	–	–	–	–	–	–	–	–	–	–	–	–	3
x_6	–	–	–	–	–	–	–	–	–	–	–	–	–	51
x_7	1	3	3	3	3	3	1	61	60	61	64	67	64	62
x_8	–	–	–	–	–	–	–	–	–	–	–	–	54	54
x_9	–	–	–	–	–	–	–	–	–	–	–	–	–	3
x_{10}	–	–	–	–	–	–	–	–	–	–	–	–	–	3
x_{11}	–	–	–	–	–	–	–	–	–	–	–	–	–	3
x_{12}	–	–	–	–	–	–	–	–	–	–	65	–	–	50
x_{13}	–	–	–	–	–	–	–	–	64	67	–	64	–	67
x_{14}	–	–	–	–	–	–	–	–	–	–	–	–	–	25
x_{15}	–	–	–	–	–	–	–	–	–	–	–	–	–	53
x_{16}	–	–	–	–	–	–	–	–	–	–	–	–	–	30
x_{17}	–	–	–	–	–	–	–	–	–	–	–	–	–	3
x_{18}	–	–	–	–	–	–	–	–	–	–	–	–	–	3
x_{19}	–	–	–	–	–	–	–	–	–	75	–	75	75	75
y	48	48	–	–	1	1	1	1060	1060	–	88	288	390	417

TABLE VIII   Good masks obtained by a depth-8 model search

Delay	Complex 4 (Q_{best} = 0.6331) Q > 0.975 Q_{best} masks								Complex 5 (Q_{best} = 0.6264) Q > 0.975 Q_{best} masks							
	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
x_1	–	–	–	–	–	–	–	–	–	–	–	–	–	–	90	86
x_2	3	3	3	3	3	3	3	3	90	92	95	94	94	95	94	93
x_3	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	4
x_4	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	4
x_5	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	4
x_6	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	70
x_7	–	3	3	3	3	3	–	3	–	92	92	92	91	91	–	89
x_8	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	48
x_9	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	3
x_{10}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	3
x_{11}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	4
x_{12}	–	–	–	–	–	–	–	–	–	–	–	89	–	–	–	70
x_{13}	–	–	–	–	–	–	–	–	–	–	81	89	–	88	82	79
x_{14}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	6
x_{15}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	34
x_{16}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	6
x_{17}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	4
x_{18}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	6
x_{19}	–	–	–	–	–	–	–	–	–	–	95	1	94	84	94	94
y	52	52	–	4	4	18	18	18	1478	1478	–	122	320	415	410	460

TABLE IX Good masks obtained by a depth-9 model search

Delay	Complex 4 ($Q_{best}=0.6340$) $Q > 0.975\ Q_{best}$ masks									Complex 5 ($Q_{best}=0.6276$) $Q > 0.975\ Q_{best}$ masks								
	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8
x_1	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	109	–	108
x_2	4	3	2	2	6	4	6	2	6	99	110	94	93	114	111	116	95	116
x_3	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
x_4	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
x_5	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	9
x_6	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	73
x_7	–	–	–	–	–	–	–	–	–	–	102	102	103	102	101	–	99	98
x_8	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
x_9	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	8
x_{10}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	9
x_{11}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
x_{12}	–	–	–	–	–	–	–	–	–	–	–	–	107	–	–	–	–	51
x_{13}	–	–	–	–	–	–	–	–	–	–	–	–	111	–	121	–	–	107
x_{14}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
x_{15}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
x_{16}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
x_{17}	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	10
x_{18}	–	–	–	–	–	–	–	–	–	–	–	127	–	–	–	–	–	10
x_{19}	–	–	–	–	–	–	–	–	–	–	–	–	–	121	–	120	120	120
y	50	50	–	2	17	1	2	22	21	1636	1636	–	95	368	94	96	520	569

model search engine.

$$m_{can} = \begin{pmatrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

This new candidate mask contains 48 "-1" elements out of 199 possible. Results of this new simulation are given in Table X.

Note that in this simulation the overall quality of the complexity 4 masks is not increased anymore with respect to the qualities obtained in previous simulations. Yet, for the complexity 5 models a slight increase in the quality of the masks is still obtained. This suggests performing another iteration, proposing a new depth-11 candidate mask (using, again, a discriminator value set to the 6%). When doing so, no increase in the quality of either the complexity-4 models or the complexity-5 models is obtained, so the iteration can be stopped. At this point, the FIR qualitative models that may best describe, in terms of quality, the system at hands are those models represented by a depth-10 mask.

This would tell the modeller to use a depth-10 FIR model (the depth-11 model is exactly the same) in which input variables x_3, x_4, x_5, x_6, x_8, x_9, x_{10}, x_{11}, x_{14}, x_{15}, x_{16}, x_{17} and x_{18} have been discarded. Only variables x_1, x_2, x_7, x_{12}, x_{13} and x_{19}, and, evidently, past values of the same output variable, are used to model the output variable. From those used variables, note that variable x_{12} only appears in 4 out of 535 possible models, and that, in fact, if another iteration were performed it would be eliminated from the depth-11 models. The new depth-11 candidate mask and the obtained results of this simulation are given in Appendix C.

So with the presented procedure not only a FIR model is found but also which is its optimal depth. Yet, it would be interesting to see how much is the computation alleviation achieved with this method. The achieved model search space simplification is computed by comparing the number of models visited when using the presented algorithm to construct depth-10 models, with the number of depth-10 models to compute using the classical FIR search method.[††] Table XI gives the number of models that have been visited in order to construct the depth-10 models presented in Table X, the number of theoretical models that should be computed for a full candidate mask of depth-10, and, finally, column three the percent of computation alleviation. The first column is calculated by adding the number of models computed, for all the allowed complexities, for depth $1-10$ as presented in this section. How to obtain the number of models that have to be computed for each number of -1 elements of the candidate mask is given in Appendix B.

[††]Which moreover implies a prior knowledge, or an heuristic decision, of the mask depth value to use.

TABLE X   Good masks obtained by a depth-10 model search

Delay	Complex 4 ($Q_{best} = 0.6340$) $Q > 0.975\,Q_{best}$										Complex 5 ($Q_{max} = 0.6312$) $Q > 0.975\,Q_{best}$									
	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9
x_1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4	4
x_2	2	2	—	—	2	2	2	—	2	2	39	42	—	—	48	47	48	—	53	45
x_3	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_5	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_7	—	—	—	—	—	—	—	—	—	—	—	37	37	40	39	37	—	38	36	36
x_8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_9	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	3
x_{10}	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_{11}	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_{12}	—	—	—	—	—	—	—	—	—	—	—	—	—	4	—	—	—	—	—	4
x_{13}	—	—	—	—	—	—	—	—	—	—	—	—	—	32	—	33	—	—	34	33
x_{14}	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_{15}	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_{16}	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_{17}	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_{18}	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4
x_{19}	—	20	—	—	—	—	—	—	—	—	—	—	33	—	33	—	7	8	47	34
y	20	20	—	—	3	—	—	10	10	3	535	535	—	—	147	—	—	137	151	148

TABLE XI   Computation alleviation achieved when using the proposed algorithm

Number of visited models $\sum_{depth=1}^{10}$ visited masks$_{depth}$	Number of models to visit using a depth 10 full candidate mask	Percent of computation alleviation
1.362.501	64,704,850	97.89%

Of course when giving the computation alleviation one must have into account the time required to study the computed FIR models, to construct the table of variable/delays used and, from it, propose a new candidate mask. These operations have been made by means of a function programmed in Matlab code which from the output text file of FIR which contains all the computed models, generates a table, for each complexity that contains the more used variables/delays. The computation time of this function is orders of magnitude lower than the FIR model computation.

3.2. Reducing the Mask Search Space Using Quantitative Data

In this second study of the same system, a spectral analysis was performed on the different trajectories of the variables. Concretely, the module of the spectral coherence function was used to find at which delays one input variable (x_i) has more energy related to the output variable (y). This second approach has been presented in "Model search space reduction from quantitative data". Table XII shows the most important delays found between the input and output variables when applying the proposed method.

With this information a candidate mask up to depth 25 can be proposed. In order to compare the obtained models with this approach, a candidate mask of depth 10 is to be proposed to FIR. This energy approach does not give information about delays 0 and 1, so the first two rows of the candidate mask should be filled with -1 elements. This would lead to a mask with $105 - 1$ elements out of 199. Although the simplification achieved is significant there still are lots of models to compute. Yet, a much wiser candidate mask could be proposed using the information given by the previous presented approach for time delays 0 and 1. So a depth-10 candidate mask can be proposed as shown below.

Note that in this candidate mask variable 18 has already been discarded. The proposed mask has 87 " -1" elements so 87 masks of complexity 2; 3,741 of complexity 3; 105,995 of complexity 4 and 2,225,895 of complexity 5 must be computed. If a full candidate mask had been proposed the number of masks to compute would have been 199, 19,701, 1,293,699 and 63,391,251 for complexities 2, 3, 4 and 5, respectively. For those masks with complexity 2 and 3, the results of this simulation are the same as simulations presented in "Reducing the mask search space using qualitative data". Table XIII shows the computation alleviation

TABLE XII   Delays to consider between inputs and output variable

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}	x_{19}	y
15	10	7	3	20	4	18	3	15	7	8	5	5	3	5	18	18	–	–	24
12	9	6		19	3	17	2	12	6	7	4	4		2	17	7			23
9	6	5		18	2	8		11	3	5		3			4	2			20
4	5	4		12		7		9	2	4		2			3				18
3	4	3		11		5		8		2					2				8
2	3	2		9		3		7											6
	2			7		2		3											5
				3				2											4
				2															3
																			2

TABLE XIII   Computation alleviation achieved when using energy considerations

Number of visited models (all considered complexities)	Number of models to visit using a depth 10 full candidate mask	Percent of computation alleviation
2,335,718	64,704,850	96,39%

achieved for the complexity 5 models whereas in Table XIV it is shown the simulation results for mask complexities 4 and 5.

$$
m_{\text{can}} = \begin{pmatrix}
-1 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & 0 & -1 & 0 & -1 & 0 & -1 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\
0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & -1 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & -1 \\
-1 & -1 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & -1 \\
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & 0 & -1 & 0 & 0 & 0 & -1 \\
-1 & -1 & -1 & 0 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & -1 & 0 & -1 & -1 & -1 & 0 & 0 & -1 \\
-1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & -1 & -1 & -1 & -1 & 0 & -1 & 0 & -1 & -1 \\
-1 & -1 & 0 & 0 & -1 & -1 & -1 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\
\end{pmatrix}
$$

From this approach it can be seen that the obtained results are very similar with those obtained in previous section (Table XI). The mask qualities obtained are quite like to those of previous simulations. In this study input variables x_1, x_2, x_6, x_7, x_{12}, x_{13} and x_{19} have been selected to model the output while all the other input variables are discarded.

3.3. Qualitative Simulation Results

In order to assess that the sub-optimal models found with the proposed algorithms are suitable to properly model the system under consideration two simulations have been performed. The first simulation uses one model of complexity 4 from those found with depth-9 in Table IX. The quality of the best model in this occasion was 0.6340. The model is used to simulate the last 500 points of the garbage incinerator output variable (NO_x gas emission), Figs. 1 and 2 show in continuous line the real NO_x data and in dotted line the simulated trajectory for this variable.

In this first simulation the used model has been:

$$y(t) = f\{y(t-1),\ y(t-4),\ x_2(t-8)\}$$

For the second simulation a complexity 5 model has been used. In this case the depth of the model is 10 and is one of the models found in Table X. The maximum quality achieved in this case was 0.6312. The model used in this simulation is:

$$y(t) = f\{y(t-1),\ y(t-7),\ x_7(t-3),\ x_2(t-8)\}$$

TABLE XIV   Results obtained when determining the most important delays in energy terms

	Complex 4 ($Q_{best} = 0.6340$) $Q > 0.975\,Q_{best}$ masks										Complex 5 ($Q_{best} = 0.6275$) $Q > 0.975\,Q_{best}$ masks									
Delay	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9
x_1	–	–	–	–	–	–	–	–	–	–	73	72	75	71	–	–	–	–	–	–
x_2	3	2	3	3	3	3	3	–	–	3	93	97	107	108	108	109	112	–	–	100
x_6	–	–	–	–	–	–	–	–	–	–	–	–	–	–	71	–	–	–	–	–
x_7	–	–	–	–	–	–	–	–	–	–	84	84	84	85	–	84	–	79	80	–
x_{12}	–	–	–	–	–	–	–	–	–	–	–	72	–	–	75	–	–	–	–	–
x_{13}	–	–	–	–	–	–	–	–	–	–	–	–	72	–	–	–	–	–	–	–
x_{19}	–	–	–	–	–	–	–	–	–	–	94	94	–	–	–	–	–	–	–	–
y	37	37	4	4	5	13	13	–	12	–	1187	1187	–	122	205	404	421	–	487	–
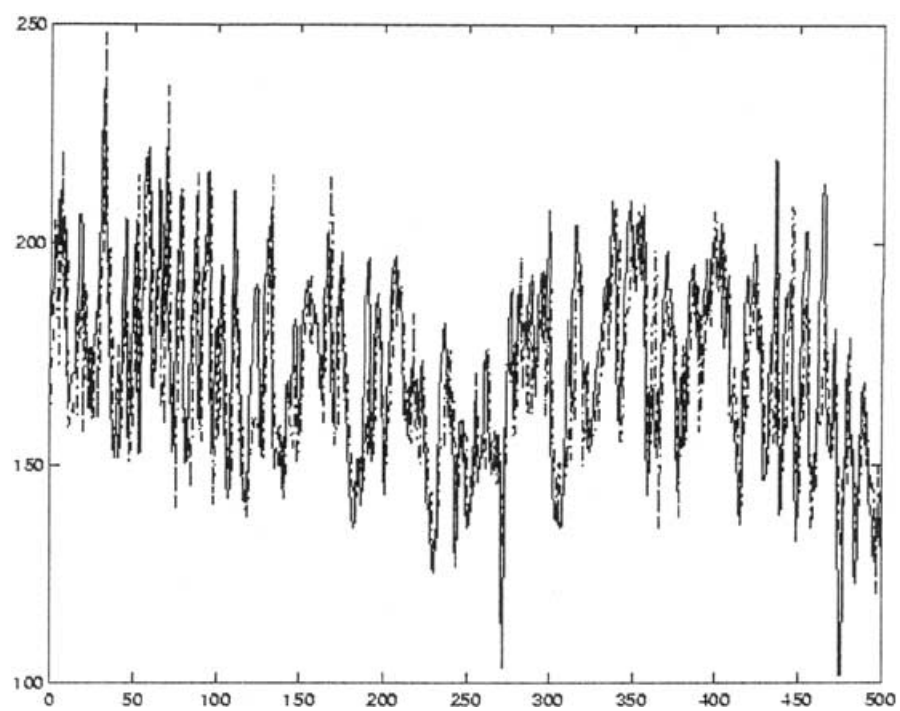
FIGURE 1   Real and simulated NO$_x$. Last 500 points.
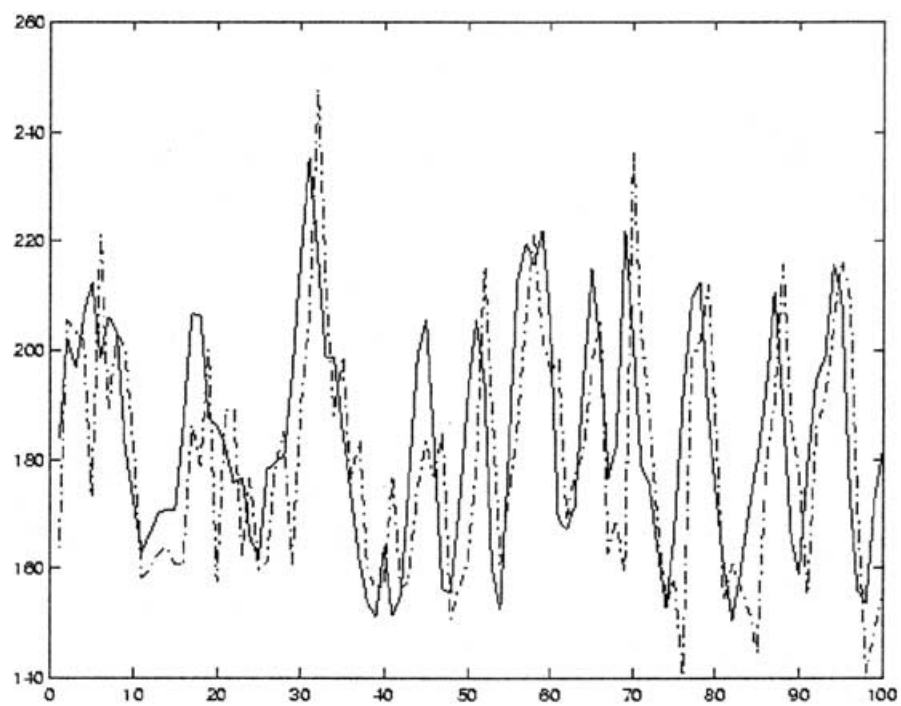
FIGURE 2   First 100 NO$_x$ FIR simulated points.
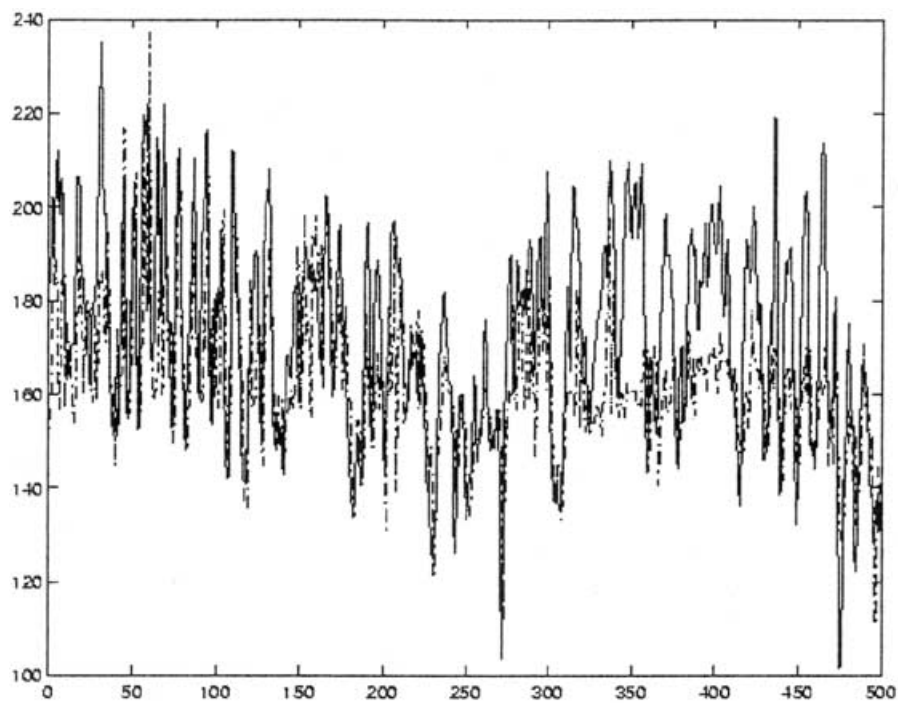
FIGURE 3   Real and simulated NO$_x$. Last 500 points.
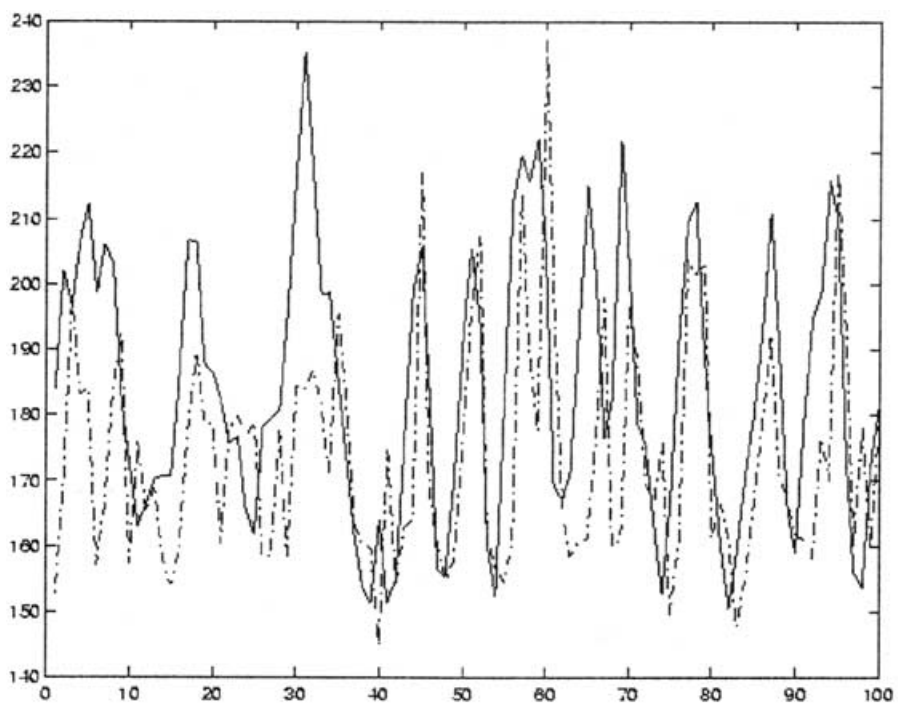
FIGURE 4   First 100 NO$_x$ FIR simulated points.

Results of this simulation are given in Figs. 3 and 4, where, again, the continuous line is for the real data and the dotted line for the simulated trajectory.

As can be stated from the figures, both simulations give quite good results in following the original NO_x trajectory. In the first performed simulation the percent average error is 1.25%; in the second case the percent average error is 3.64% (average computed with all the 500 points simulated).

4. CONCLUSIONS

Two approaches have been presented to tackle the problem of modelling a large-scale system with the FIR methodology, a previous unmanageable task. Both approaches reduce the model search space of the Fuzzy Inductive Reasoning Qualitative Modelling Engine.

The first of these approaches makes use of the qualitative data of the system to achieve computation alleviation. So the presented algorithm may be viewed as an intermediate module between the Fuzzification Module and the Qualitative Modelling Engine of FIR which receives feedback from the QME. At each step of the algorithm a new deeper candidate mask is proposed to FIR which only has some of its elements equal to -1. How these elements are set to 0 or -1 depends on information about the best qualitative models found in the previous model search. Using this procedure, a good computation reduction/model quality compromise is obtained.

The second approach deals directly with the real variable trajectories gathered up from the system, i.e. their real episodes. It can be viewed as a pre-processing unit, previous to the fuzzyfication module, which allows to gather information about which are the most important input variables, at given delays, related with the output. In this approach each variable trajectory is considered as the realisation of a stochastic process, i.e. the measure of a physical variable given by a (noisy) sensor in a possible noisy environment. The spectral coherence function is computed for each pair of input–output trajectories and from it, the most important delays are deduced. Although, only information from 2 to ∞ delays can be gathered, an important computation reduction is achieved proposing FIR a unique sparse candidate mask.

The second approach can be combined with the first one, obtaining, with the latter, a sparse depth 2 candidate mask, and with the former, information about delays from 2 to ∞, leading, together, to a unique sparse candidate mask. This operation has been performed in "Reducing the mask search space using quantitative data", achieving a considerable reduction on the FIR model computation cost.

References

de Albornoz, A., (1996) "Inductive reasoning and reconstruction analysis: two complementary tools for qualitative fault monitoring of large-scale systems" Llenguatges i Sistemes Informàtics, Ph.D. dissertation, Universitat Politècnica de Catalunya (Barcelona, Spain).

Carvajal, R. and Nebot, A. (1997) "Growth model for white shrimp in semi-intensive farming using inductive reasoning methodology", *Comput. Electronics Agric.* **19**, 187–210.

Cellier, F.E. (1991) *Continuous System Modelling* (Springer, New York).

Cellier, F.E., Lòpez, J., Nebot, A. and Cembrano, G. (1996). "Means for estimating the forecasting error in a fuzzy inductive reasoning", *Proc. ESM'96, European Simulation Multiconference*, Budapest, Hungary, pp. 654–660.

Jerez, A. and Nebot, A. (1997). "Genetic algorithms vs. Classical search techniques for identification of fuzzy models", *Proc. EUFIT'97, 5th European Congress on Intelligent Techniques and Soft Computing*, Aachen, Germany, 8–12 September, pp. 769–773.

Kalouptsidis, N. (1997) *Signal Processing Systems, Theory and Design* (Wiley-Interscience, New York).

López, J., (1999) "Time series prediction using inductive reasoning techniques" Organització i Control de Sistemes Industrials, Ph.D. dissertation Universitat Politècnica de Catalunya (Barcelona, Spain).

Marple, S.L. (1987) *Digital Spectral Analysis with Applications* (Prentice-Hall, Engelwood Cliffs, NJ).

Mirats i Tur, J.M., Huber, R.M., Cellier, F.E. and Qin, S.J. (2002) "On the selection of variables for systems modelling and control", *Int. J. Gen. Syst.*, In press.

Nebot, A. and Jerez, A. (1997). "Assessment of classical search techniques for identification of fuzzy models", *Proc. EUFIT'97, 5th European Congress on Intelligent Techniques and Soft Computing*, Aachen, Germany, 8–12 September, pp. 904–909.

Nebot, A., Valdés, J.J., Guiot, M., Alquézar, R. and Vallverdú, M. (1998). "Fuzzy inductive reasoning approaches to the identification of models of the central nervous system control", *Proc. EIS'98, International Symposium on Engineering of Intelligent Systems*, Tenerife, Spain, 11–13 February, pp. 190–196.

Valle, S., Li, Weihua and Qin, S.J. (1999) "Selection of the number of principal components: the variance of the reconstruction error criterion with a comparison to other methods", *Ind. Eng. Chem. Res.* **38**, 4389–4401.

APPENDIX A

The process analysed in this work is a thermal incinerator. The description given here is a general description of the functioning of these kinds of systems (Valle *et al.*, 1999). In this unit, high temperature and chemical reactions burn up process fumes, and change them into harmless carbon dioxide and water vapours, which are then released through a stack into the atmosphere. The fumes processed in the incinerator are commonly called NO_x fumes, and mainly consist of: nitric oxide (NO), nitrogen dioxide (NO_2), and nitrous oxide (N_2O). Hydrogen cyanide (HCN) is also present. The gases are converted into nitrogen and water vapour in a three stages combustion process. The first stage is reduction, the second is re-oxidation, and the third is catalytic oxidation. A de-mineralised water heat exchanger is located between the second and third combustion stages (Fig. A1).

   Data were gathered up from the system at a sampling rate of 1 min and 43,200 data points were recorded. The considered output variable for this study is the emission of NO_x gas. Table A1 gives a short description of the variables in the system that this paper deals with. Fig. A1 shows a general diagram describing the layout of the system.
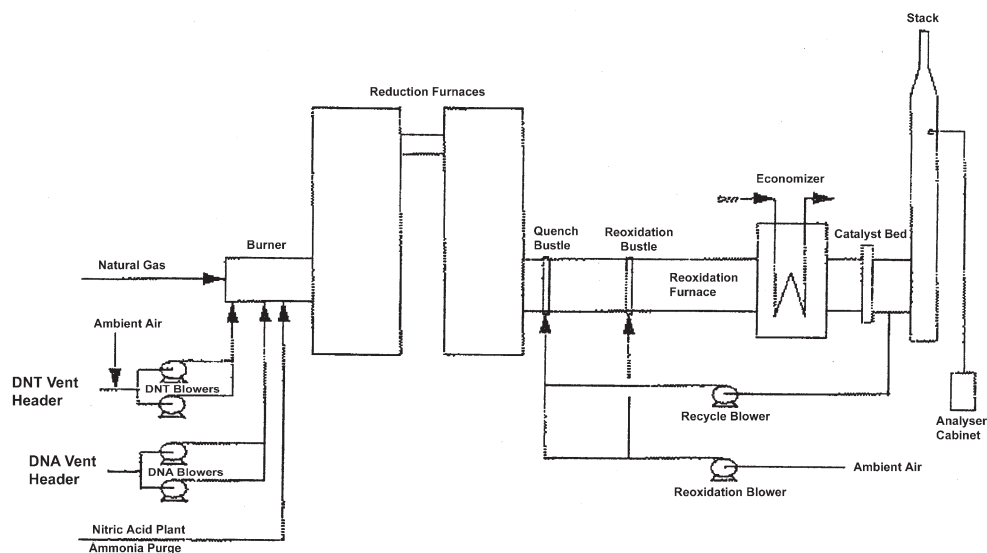

FIGURE A1   Incineration process scheme.

TABLE AI Variable description in the garbage incinerator plant

Variable	Physical meaning
1 (input)	DNT unit one vent (SCFM)
2 (input)	DNT unit two vent (SCFM)
3 (input)	A column overhead (F)
4 (input)	C column overhead (F)
5 (input)	DNT vent rate (SCFM)
6 (input)	Stack gas recycle (ACFM)
7 (input)	Stack gas recycle (F)
8 (input)	Strip vent rate (SCFM)
9 (input)	DNT vent head VA (WC)
10 (input)	Column A top pressure (inch W)
11 (input)	Column B top pressure (inch W)
13 (input)	Column D top pressure (inch W)
14 (input)	DP on A column (PSIG)
15 (input)	DP on B column (PSIG)
16 (input)	DP on C column (PSIG)
17 (input)	DP on D column (PSIG)
18 (input)	Excess O_2 in stack (%)
19 (input)	Reduction furnace (F)
20 (output)	NO_x, (PPM)

The reduction section is a large natural gas (methane) furnace. The air for the combustion is provided by the vent header (DNT), or stripper vent header (DNA). The burning of natural gas provides all the heat required for the reduction stage. There are four major chemical reactions that occur in the reduction section furnace:
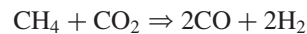
1. Burning of natural gas

$$CH_4 + 2O_2 \Rightarrow 2H_2O + CO_2$$
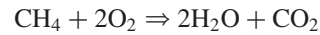
2. NO_x gas is destroyed by

$$CH_4 + 2NO_2 \Rightarrow N_2 + 2H_2O + CO_2$$

3. Additional fuel is added without air to increase the reaction between natural gas and the NO_x, and to remove any trace of oxygen in the system

$$CH_4 + 4NO \Rightarrow 2N_2 + 2H_2O + CO_2$$

4. CO_2 gas is destroyed by

$$CH_4 + CO_2 \Rightarrow 2CO + 2H_2$$

The hot gases leaving the reduction furnace thus consist of: carbon monoxide, nitrogen, hydrogen, water vapour, carbon dioxide, NO_x (below 200 ppm), and HCN (below 500 ppm). After the reduction furnace hot gases are quenched by mixing them with cooler recycled stack gas. The quenched gas then flows into the stage section called the re-oxidation stage. This section is not used for NO_x abatement, but to convert CO and H_2, the combustibles from the reduction furnace into CO_2 and water vapour. In this section fuel gas is not used, and a

blower adds ambient air to the section. Two additional reactions occur:

$$2CO + 2O_2 \Rightarrow O_2 + 2CO_2 \quad 2H_2 + O_2 \Rightarrow 2H_2O$$

The hot gases from the re-oxidation section then flow through an economiser where they are cooled by de-mineralising water. The hot de-mineralised water (DMW) is then routed to the plant. Gases exiting the economiser flow through a honeycomb grid of platinum catalyst, where the CO and the organics are converted to inert flue gases before their discharge from the stack to the atmosphere:

$$CO + 2HCN + C_6H_5CH_3 + O_2 \Rightarrow CO_2 + H_2O + N_2$$
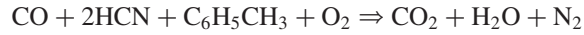
Analyser probes in the stack monitor the oxygen, CO, CO_2, and NO_x levels.

APPENDIX B

There are two main reasons that increase the computation time of the FIR models, one is the number of masks to compute and the other is the data matrix size. In this paper, the data matrices used to compute the masks are quite big $(36,720 \times 20)$ so the process of calculating masks is quite slow. All the simulations presented in this paper were performed in an Ultra Sparc II Workstation running sunOs 5.7.

The total number of models that have to be computed when a mask candidate with n elements set to " -1 " and the maximum allowed complexity is c, is given by the formula:

$$\begin{bmatrix} n \\ c \end{bmatrix} \equiv \begin{bmatrix} n \\ 1 \end{bmatrix} + \begin{bmatrix} n \\ 2 \end{bmatrix} + \cdots + \begin{bmatrix} n \\ c-1 \end{bmatrix}$$

For example, when a full depth-10 candidate mask is used, there are 199 possible -1 elements. If the maximum allowed complexity is 5, the number of models to compute is:

$$\begin{bmatrix} 199 \\ 5 \end{bmatrix} \equiv \begin{bmatrix} 199 \\ 1 \end{bmatrix} + \begin{bmatrix} 199 \\ 2 \end{bmatrix} + \begin{bmatrix} 199 \\ 3 \end{bmatrix} + \begin{bmatrix} 199 \\ 4 \end{bmatrix}$$

$$= 199 + 19,701 + 1,293,699 + 63,391,251 = 64,704,850$$

There are 199 models of complexity 2; 19,701 models of complexity 3, and so on, leading to a total number of models of almost 65 millions.

APPENDIX C

The depth mask -11 used in the last step of the algorithm is presented in this Appendix. This new mask candidate can be derived directly from Table X, using, again, a discriminator value

set to the 6%.

$$
m_{\text{can}} = \begin{pmatrix}
-1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
-1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}
$$

In this mask there are 49 " − 1" elements out of 119 possible. The obtained results are shown in Table XII. For this simulation the qualities of the masks no longer increased, they stayed at the same value than in the previous simulation. No more tables are constructed because no new information would be added.

Just to assess the results obtained with the given algorithm some more simulations, up to depth 16, were performed. No increase in the mask qualities was obtained in any of those simulations, only a little fluctuation of the complexity-5 model quality was observed in the depth-15 models. Finally, Fig. C1 shows the behaviour of the mask quality versus the depth
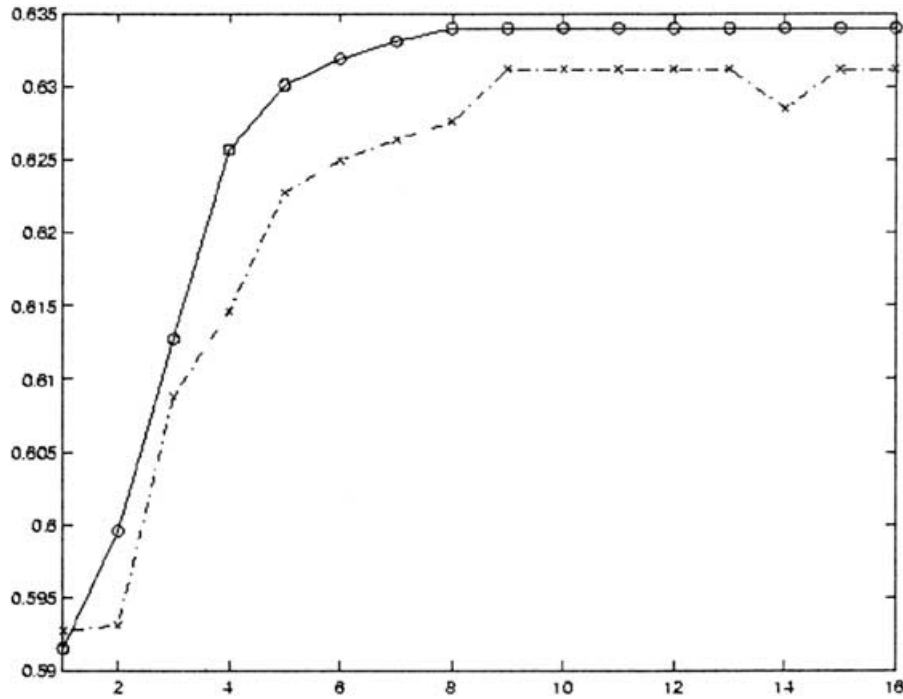

FIGURE C1   Quality versus depth of the mask.

of the mask. Continuous line is for complexity 4 and dotted line for complexity 5 masks. Those qualities for the one depth masks are not represented in order the y-axis scale o the figure to be clearer.

Josep M. Mirats i Tur received his title of Enginyer de Telecommunicacions (Electrical Engineering, specialised in electronics) in 1995, from the Universitat Politècnica de Catalunya (UPC). He finished his Ph.D on qualitative modelling in the Institute of Robotics depending of both the UPC and CSIC, Centro Superior de investigaciones científicas (Scientist research Spanish council) in November 2001. Before joining the Institute he was working for the private industry in the research department of the Seat-Volkswagen Company. He has been involved as research support engineer within the Institute for different European and CICYT (Comisión Interministerial de Ciencia y Tecnología) projects.His main scientific interests concerns simplifying the computation cost inherent to the existent qualitative modelling and simulation methodologies, concretely with the FIR methodology, and use it to model and simulate large-scale systems.

François E. Cellier received his B.S. degree in Electrical Engineering from the Swiss Federal Institute of Technology (ETH) Zürich in 1972, his M.S. degree in Automatic Control in 1973, and his Ph.D. degree in Technical Sciences in 1979, all from the same university. Dr Cellier joined the University of Arizona in 1984 as Associate Professor. His main scientific interests concern modeling and simulation methodologies, and the design of advanced software systems for simulation, computer-aided modelling, and computer-aided design. Dr Cellier has authored or co-authored more than eighty technical publications, and he has edited four books. He recently published his first textbook on Continuous System Modeling (Springer-Verlag: New York, 1991). He served as General Chairman or Program Chairman of many international conferences, most recently ICBGM'93 (SCS International Conference on Bond Graph Modeling, San Diego, January 1993), CACSD'94 (IEEE/IFAC Symposium on Computer-Aided Control System Design, Tucson, March 1994), ICQFN'94 (SCS International Conference on Qualitative Information, Fuzzy Techniques, and Neural Networks in Simulation, Barcelona, June 1994), ICBGM'95 (Las Vegas, January 1995), WMC'96 (SCS Western Simulation MultiConference, San Diego, January 1996), WMC'97 (Tucson, January 1997). He is Associate Editor of several simulation related journals, and he served as vice-chairman on two committees for standardisation of simulation and modeling software. Dr Cellier was promoted to the rank of Full Professor in 1997.

Rafael M. Huber received his Ingeniero Industrial (Electrical Engineering branch) and his Ph.D. in Ingeniería Industrial in 1976, both from the Universitat Politècnica de Catalunya (UPC). His present position is Catedrático de Universidad (Professor) at the Automatic Control Department of the UPC and nowadays he is serving as director of the Instituto de Robótica e Informática Industrial (IRI) depending of the UPC and the Spanish Council of Scientific Research (CSIC). His main scientific interests concern modelling and simulation methodology and the design of advanced simulation

environments. Its present research focus qualitative modelling and simulation and its application to dynamic systems fault detection and diagnosis. He has been involved as research engineer or research head in projects with the spanish industry, the Comisión Interministerial de Ciencia y Tecnología (CICYT), the CSIC, the European Space Agency and the U.S. National Science Foundation. Prof. Huber has authored or co-authored more than forty technical publications and edited two books related with continuous system modelling.