Neural networks and their applications, J. Hérault, ed., 1994.

FROM KNOWLEDGE-BASED MODELS TO RECURRENT NETWORKS: AN APPLICATION TO AN INDUSTRIAL DISTILLATION PROCESS

Jean-Luc PLOIX
NETRAL
14, rue Verdi
92140 ISSY LES MOULINEAUX, France
(1) 46 38 75 152

Gérard DREYFUS
ESPCI, Laboratoire d'Electronique
10, Rue Vauquelin
75005 PARIS, France
(1) 40 79 45 41

Jean-Pierre CORRIOU
LSGC-CNRS-ENSIC
1, rue Grandville - BP 451
54001 NANCY Cedex, France
(16) 83 17 52 13

Dominique PASCAL
ELF ATOCHEM
Route de la Chimie, BP 86
76700 HARFLEUR, France
(16) 35 13 25 28

Abstract

A real-time distillation column simulator for early fault detection has been designed using an original methodology which capitalizes on the wealth of mathematical knowledge available on the distillation process, while retaining the flexibility due to the non-linear modeling abilities of neural networks. The resulting recurrent neural network has 102 state variables (probably the highest-order dynamical network ever trained and implemented for industrial purposes), over one thousand neurons, but a small number of trainable weights. It achieves the required accuracy with real-time operation on a PC, whereas conventional dynamical simulators experience difficulties when faced with real-time requirements.

Résumé

Un simulateur de colonne à distiller, destiné à la détection précoce de pannes, a été conçu en utilisant une méthodologie originale qui met à profit toute la richesse des connaissances mathématiques disponibles sur le processus, et qui conserve néanmoins la souplesse rendue possible par l'aptitude des réseaux de neurones à modéliser des systèmes non linéaires. Le réseau bouclé ainsi conçu comprend 102 variables d'état (c'est probablement le réseau bouclé d'ordre le plus élevé jamais utilisé dans une application industrielle), plus de mille neurones, mais un petit nombre de poids ajustables. La précision requise est obtenue en temps réel sur un PC, alors que les simulateurs conventionnels ont du mal à affronter les contraintes du temps réel.

1. Introduction

The detection of faults in complex systems is an important problem in large-scale processes such as are often found in the chemical industry. The present paper describes a simulator designed for early fault detection in a distillation column which is an element of a steam cracking unit

A typical distillation column process about one hundred tons of material per hour; the purity of the resulting products must comply with severe specifications irrespective of the quality of the feed. One problem commonly encountered in such a process is flooding, a malfunction due to hydrodynamic instabilities, which results in the column ceasing to distillate within a few hours after inception of the phenomenon; the recovery time after detecting the fault and taking the proper corrective steps is also on the order of hours; therefore, early detection of the phenomenon is of utmost importance. After the phenomenon has occurred, experts are usually able to detect a few hints from the evolution of the measurements which are taken from the column. However, these elements are usually very difficult to detect in real time because of noise and ambiguities.

In order to overcome these difficulties, essentially three approaches can be considered :

(i) use an expert system which would combine the data from the measurements of the column in order to diagnose the faults; given the noise and variability of the measurements, this seems to be impractical;

(ii) use a statistical approach; two possibilities arise :

- classifying the measurement curves into "normal operation" and "abnormal operation" classes ; one of the problems encoutered with this approach is the fact that the number of examples of abnormal operation is small;
- modeling the normal operation of the column, and detecting a malfunction through the occurrence of a statistically significant difference between the predictions of the model and the measurements made on the column; this approach was chosen because there is a wealth of mathematical knowledge and of experimental data on the normal operation of the column, which can be taken advantage of.

The first part of the paper describes the fundamentals of the distillation process. The second part describes the methodology used for the design and training of a recurrent neural network which simulates the normal operation of the process. Finally, representative results are presented and perspectives are outlined.

1. A crash course on distillation

Figure 1 shows a schematic diagram of the distillation column under consideration. Distillation occurs through successive mass and heat exchanges [Rovaglio et al. 1990] on 50 different trays. The raw material is fed at an intermediate level, the distillate (rich in volatile components) is obtained beyond the condenser situated at the top of the column and the residue (rich in heavy components) is extracted from the boiler at the bottom. The measurements taken on-site are the temperatures of eight trays in the column as indicated on the diagram, and the mole fractions of the components at the top and at the bottom of the column. Therefore, there are only 10 measurable variables.

The column under consideration processes a mixture of three main species, with impurities up to 9 %; since all mole fractions must sum to one, two mole fractions only are independent variables; we denote by $x_{a,i}$ (resp. $x_{b,i}$) the mole fraction of component a (resp. component b) in the liquid at tray i, and by $y_{a,i}$ (resp. $y_{b,i}$) the mole fraction of component a (resp. component b) in the vapour at tray i.

Figure 2 shows schematically the phenomena occurring at tray i: the dynamics of distillation at each tray can be described by state equations involving two variables ($x_{a,i}$ and $x_{b,i}$) per tray. Therefore, the dynamics of the whole system of trays involves 100 variables. We denote by V and L the internal flows of vapour and of liquid, which we consider as being approximately independent of i. The mass balances can be written as follows :

$$M_i \dot{x}_{a,i} = L\, x_{a,i-1} + V\, y_{a,i+1} - L\, x_{a,i} - V\, y_{a,i}$$
$$M_i \dot{x}_{b,i} = L\, x_{b,i-1} + V\, y_{b,i+1} - L\, x_{a,i} - V\, y_{b,i} \qquad (1)$$

These equations are linear with respect to the mole fractions, but the mole fractions in the liquid and in the vapour phases are related by non linear equilibrium equations resulting from thermodynamics:

$$y_{a,i} = G_a(P, x_{a,i}, x_{b,i}) \text{ and } y_{b,i} = G_b(P, x_{a,i}, x_{b,i}) , \qquad (2)$$

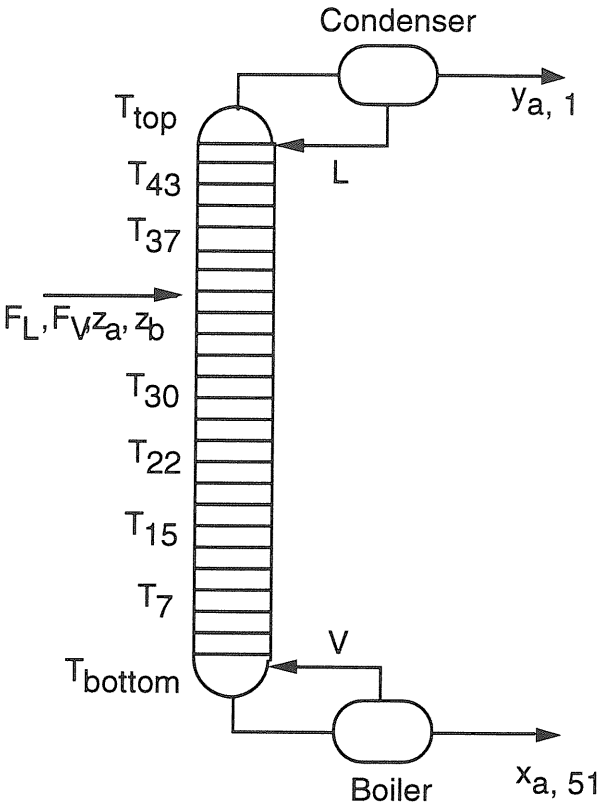

Figure 1

z_a, z_b are the mole fractions of a and b in the raw material, F_L and F_V are the liquid and vapour flows in the feed.

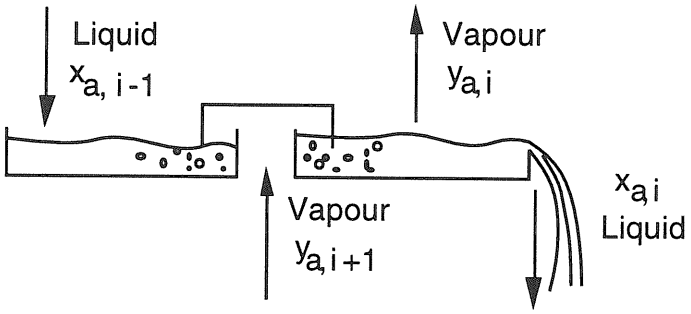

Figure 2

where P is the pressure.

Combining (1) and (2) and discretizing the differential equations yield the non-linear difference equations of each tray:

$$x_{a,i}(t+1) = F_a[P, L, V, x_{a,i}(t), x_{b,i}(t), x_{a,i-1}(t), x_{b,i-1}(t), x_{a,i+1}(t), x_{b,i+1}(t)]$$
$$x_{b,i}(t+1) = F_b[P, L, V, x_{a,i}(t), x_{b,i}(t), x_{a,i-1}(t), x_{b,i-1}(t), x_{a,i+1}(t), x_{b,i+1}(t)] \quad (3)$$

As mentioned above, eight measured variables out of ten are temperatures, not mole fractions. Therefore, one has to take into account the non-linear relations between the mole fractions and the temperature [Lang et al. 1990]:

$$T_i = \Theta(P, x_{a,i}, x_{b,i}) \quad (4$$

In addition to the above equations, the feeding of the column can be described as one additional (fictitious) tray.

Therefore, the dynamics of the whole column can be described, as a first approximation, by a set of 51 coupled non-linear difference equations, each of them involving 2 state variables, thereby combining into a model with 102 state variables. These equations could be solved numerically, but they are not accurate enough for the model to be used for fault diagnosis. The required accuracy necessitates the use of much more complex models, which cannot be solved in real time on personal computers. On the other hand, black-box identification of such a process with a neural network would be a huge and probably hopeless task, and would just throw away a lot a useful and relevant information. Therefore, we have combined the knowledge involved in the above model with the

traditional black-box approach of neural networks in order to obtain the required accuracy without increasing substantially the computation time.

2. The design and training of the recurrent neural network

The basic idea underlying the methodology is the following: a neural network is designed, which is governed by the equations of the model described above, so as to account for the basic dynamical properties of the process; the required accuracy is achieved by tuning the parameters to the measurements through training, and by using additional small, local black boxes which take into account the phenomena which are not described by the knowledge model.

Each tray was modeled by two small neural networks (3 hidden neurons each) implementing the right-hand side of non-linear equations (3), and these networks are stacked (with shared weights) so as to build a complete model of the column, as shown on Figure 3. The resulting canonical form [Nerrand et al. 1993] of the recurrent network is shown on Figure 4; it has 7 inputs, 10 outputs, 2 of which are state variables and 8 of which are related to state variables by non-linear relations (4). No measurement can be performed on the other state variables. Due to weight sharing, and to many weights being fixed by the physics of the process, the network has 32 adjustable weights only.

Algorithms used for training recurrent networks for dynamical process identification, when some state variables are not measured (thus cannot be assigned desired values during training), are described in [Nerrand et al. 1993a] and in [Nerrand et al. 1994]. In the present case, a two-step training procedure was used. :

(i) a training set was generated by using a commercial simulation package (PROSIM) which solves the exact equations of the complete knowledge model; after completion of training, the neural network was able to duplicate the results of the knowledge-based model, which, as mentioned above, was not accurate enough for the purpose of fault detection; thus, eight small black-box neural networks were appended to the first model; they compute corrections to the predicted temperatures, in order to produce accurate predictions of the measured temperatures;

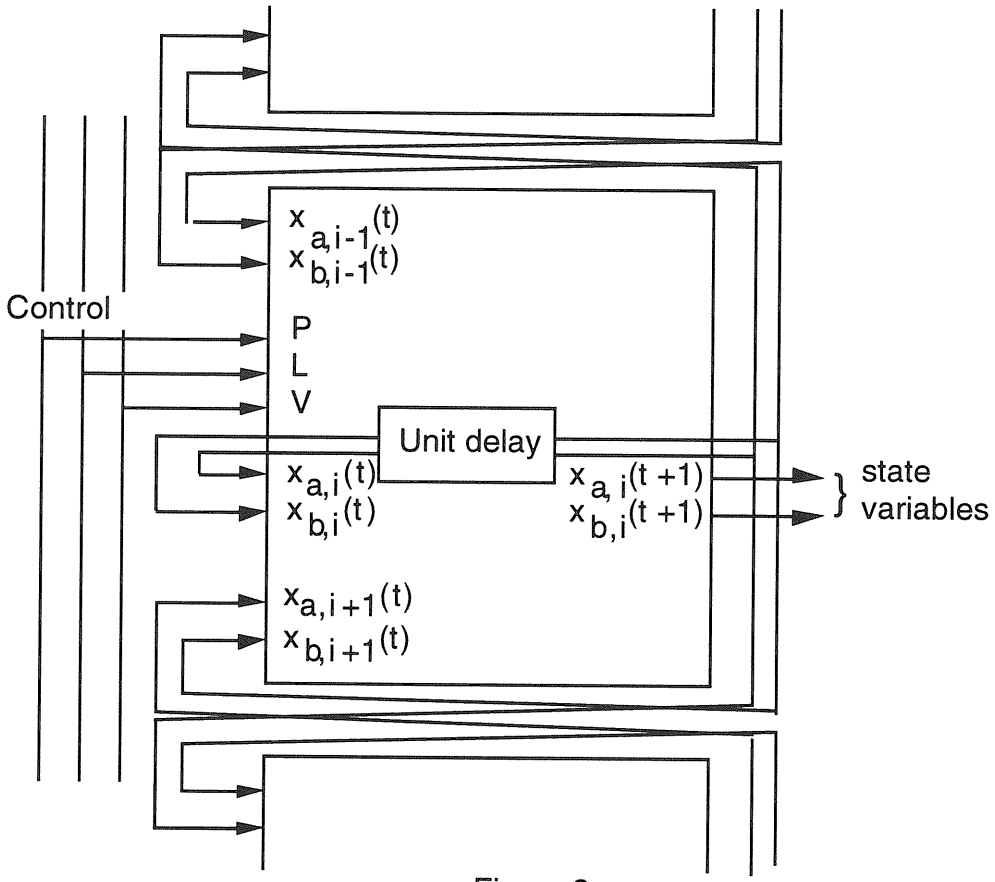

Figure 3
Modular construction of the neural network

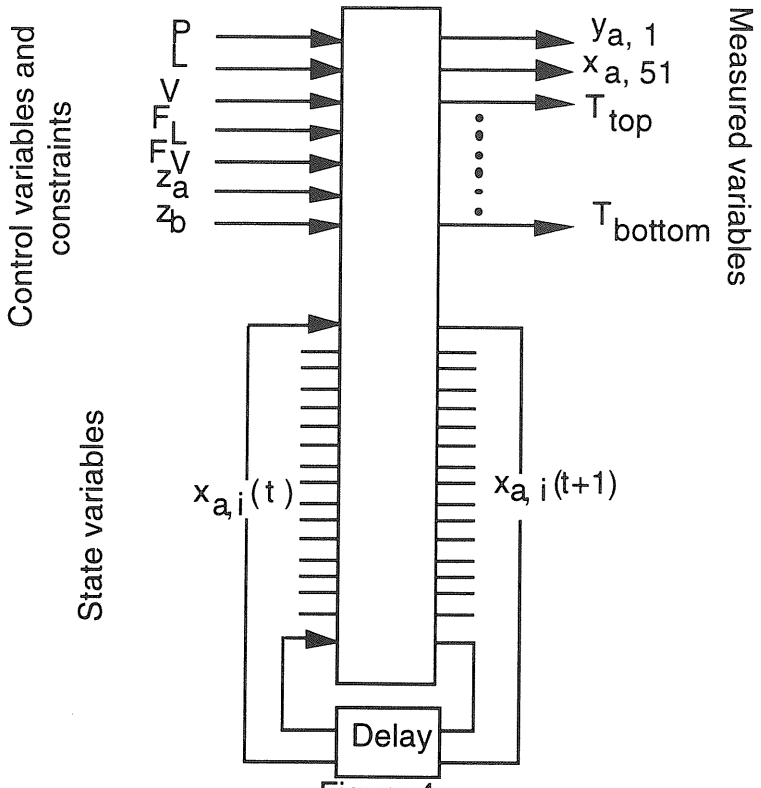

Figure 4
Overall network

(ii) during the second step, the training set consisted of measurements, performed on the column during periods of normal operation, of the eight temperatures and of the two mole fractions.

3. Results

One of the industrial requirements of the project was that the simulator should run on a PC. The neural network was developed and trained on a 486DX PC at 66Mhz clock rate. Under these conditions, the simulator actually runs faster than real-time: one minute of computation is enough for modeling 6 minutes of real operation.

The eight temperatures and two concentrations measured on the distillation column are sampled every three minutes. Figure 5 shows the measured temperature at the top of the column during 50 hours (in January 1994), and the predicted temperature at the end of the first training. Note that (i) as mentioned above, the desired values during this preliminary training are not the measured values, but the values computed by the knowledge-based model, and that (ii) one point out of six only is used for training, the intermediate points being part of the test set. The curves show that, at this stage, the dynamics of the column is essentially correct, but there is a discrepancy between the predicted temperature and the measured temperature, which precludes the use of this model for fault detection.

Figure 6 shows the data pertaining to the same time period after appending the corrector networks and training with respect to the measured temperatures. It is clear that the black-box part of the network has improved the results to a large extent.

Figure 7 shows results obtained one month later. The neural network predictions are very accurate in the considered range of operation, except during a few hours starting around t = 72 hours, where a strong deviation arises. At that time, a typical malfunction of the column occurred; a corrective action was taken by the operators roughly 90 minutes after the inception of the malfunction, and the column came back to normal condition after three hours. The curves show that, if the neural network predictor had been available to the operators, the latter would have taken the corrective action as soon as the discrepancy became significantly larger than the variance of the prediction error, i.e. approximately 20 minutes after the inception of malfunction. In validation operation, the

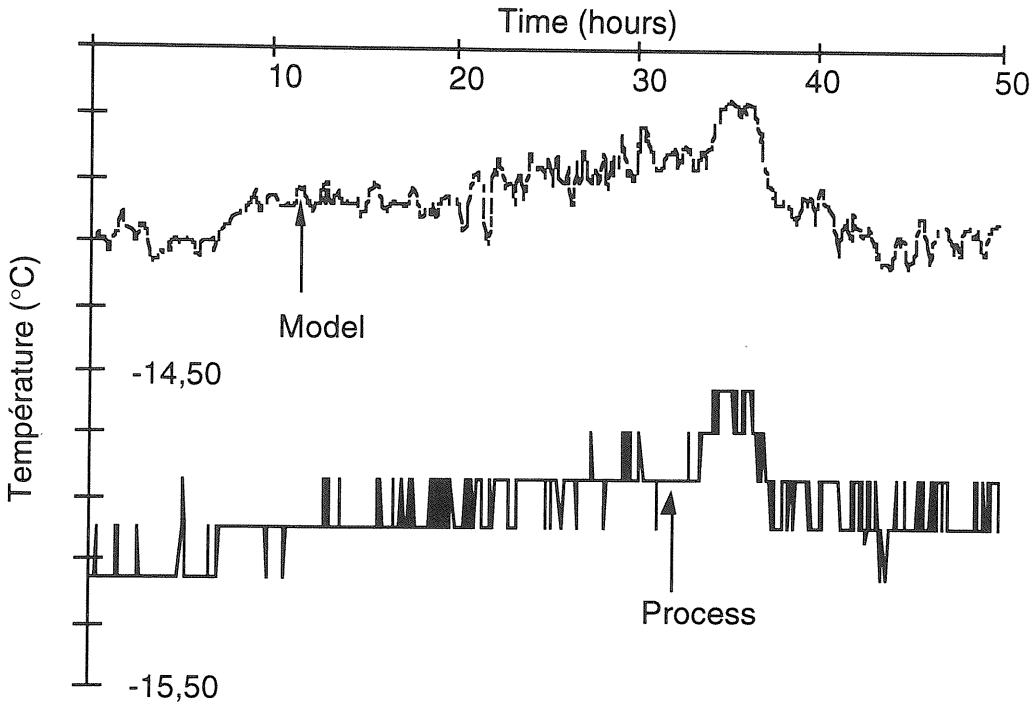

Figure 5
Typical results obtained on the training period after the first training.
One point out of six is used for training, intermediate points are part of the test set

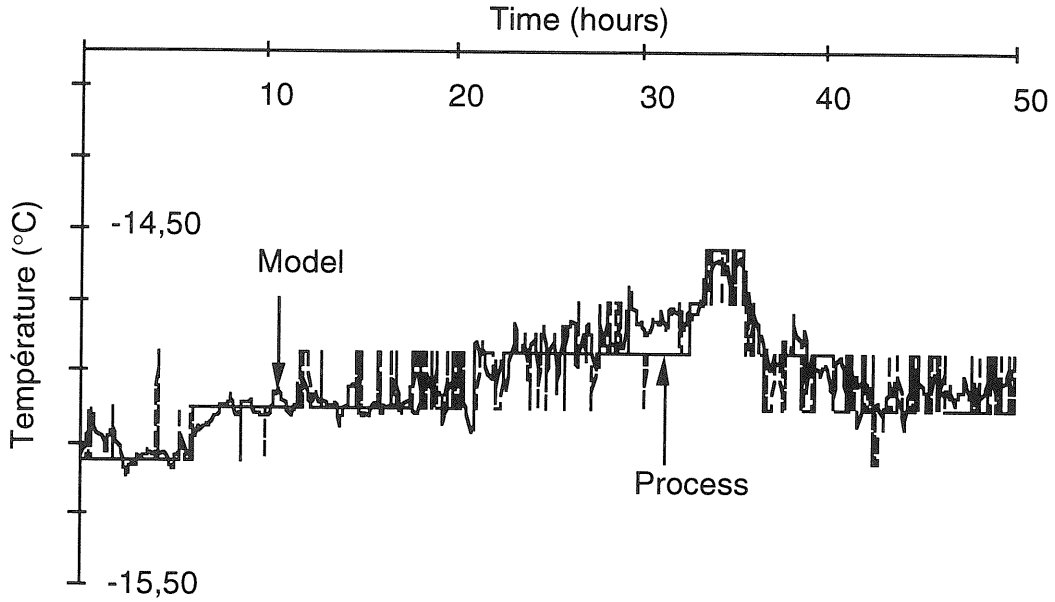

Figure 6
Typical results obtained on the training period after the second training.
One point out of six is used for training, intermediate points are part of the test set

simulator will compute on-line a 90% confidence interval for the prediction error, and will sound an alarm when the latter leaves the confidence interval.

4. Conclusion and perspectives

During, the past few years, one of the groups involved in this project devoted a lot of effort to developing efficient techniques for training recurrent networks, with special emphasis on process modeling [Nerrand et al. 1993a], [Nerrand et al. 1994] and control [Rivals et al. 1993]. The results presented here show that it is indeed possible to use recurrent neural networks in an industrial context, for the modeling of large scale, complex dynamical

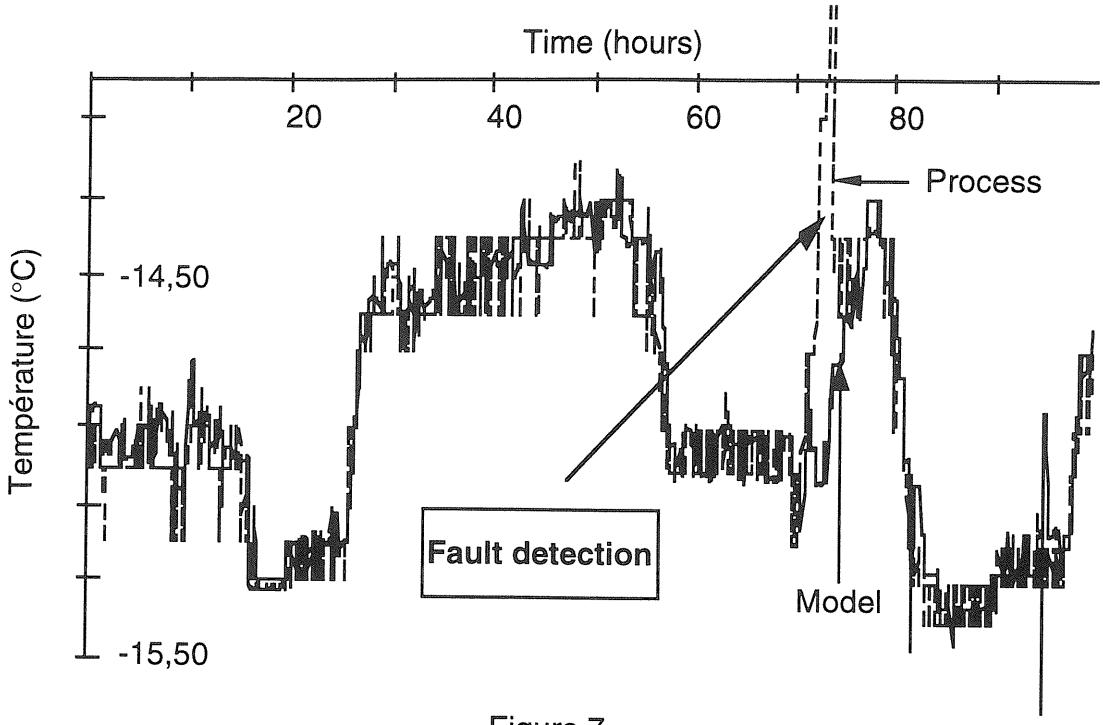

Figure 7
Typical results obtained one month after training,
showing the detection of a fault

systems, provided a methodology is used, which allows to take advantage of all the available mathematical information, albeit approximate. Such a method leads to a satisfactory accuracy, without substantially increasing the computation time with respect to knowledge-based models. In the present case, it has allowed us to build a distillation column simulator operating in real time on a PC, whereas a knowledge-based simulator of comparable accuracy would have been much more computationally demanding.

Fault detection is not the only use of such a simulator. Since it is fast and accurate, it can be used as a pedagogical tool for operator training. It can also be used in an overall control architecture; these projects are in progress. Finally, it should be mentioned that the modular construction of the neural network model makes its extension to other types of distillation columns relatively straightforward.

Acknowledgements: the authors are grateful to L. Personnaz, P. Roussel-Ragot and O. Nerrand for helpful discussions.

References:

Lang L., Gilles E.D. (1990). **Nonlinear Observers for Distillation Columns**. Computers Chem. Engng. vol. 14, pp.1297-1301.

Nerrand O., Roussel-Ragot P., Personnaz L., Dreyfus G., Marcos S. (1993). **Neural Networks and Non-linear Adaptive Filtering: Unifying Concepts and New Algorithms.** Neural Computation vol. 5, pp. 165-197.

Nerrand O., Urbani D., Roussel-Ragot P., Personnaz L., Dreyfus G. (1994) **Training Recurrent Neural Networks: Why and How? An Illustration in Process Modeling**. IEEE Transactions on Neural Networks vol.5, pp. 178-184, (1994).

Rivals I., Canas D., Personnaz L., Dreyfus G.(1994). **Modeling and Control of Mobile Robots and Intelligent Vehicles by Neural Networks**, IEEE Conference on Intelligent Vehicles (Paris).

Rovaglio M., Ranzi E., Biardi G., Faravelli T. (1990). **Rigorous Dynamics and Control of Continuous Distillation Systems : Simulation and Experimental Results.** Computers Chem. Engng. vol. 14, pp. 871-887.