

EARLY FAULT DETECTION IN A DISTILLATION COLUMN: AN INDUSTRIAL APPLICATION OF KNOWLEDGE-BASED NEURAL MODELLING

J.L. PLOIX

NETRAL S.A.

14, rue Verdi

92130 ISSY LÈS MOULINEAUX

G. DREYFUS

École Supérieure de Physique et de Chimie Industrielles

Laboratoire d'Électronique

10, rue Vauquelin

75005 PARIS

Gerard.Dreyfus@espci.fr

1 - INTRODUCTION

One of the most widespread misconceptions about neural networks is the fact that they are "black boxes" which (i) do not make any use of prior knowledge of the process to be modelled, and (ii) cannot be "understood" by the expert of the process. We show that, on the contrary, neural networks can be used as "grey box models", and that the designer can take full advantage of the mathematical knowledge which may exist on the process. Using a knowledge-based neural model, we have been able to design a real-time distillation column simulator, implemented on a PC, which allows the early detection of faults. The neural network is a dynamic model (recurrent neural net) with 102 state variables, presumably the largest recurrent neural network ever trained and implemented for industrial purposes.

2 - THE PROBLEM

2.1 - Objectives and strategy

A typical distillation column processes about one hundred tons of material per hour, complying with severe specifications on the purity of the distillate, irrespective of the quality of the raw product. Because of the very long time constants involved in the operation of such processes, the early detection of faults is an important problem: a malfunction which is not detected very shortly after its inception may result in wasting tens of hours of operation. The difficulty of automatic fault detection arises from the fact that the number of measured variables is usually small as compared to the number of state variables, that they are noisy, and that trends observed in the measurements are often ambiguous. The noise and variability of the measurements essentially preclude the use of expert systems in real time. By contrast, statistical approaches are good candidates for contributing to the solution of the problem.

One possible statistical approach, frequently encountered in the process industry, is classification: the curves resulting from the measurements taken on the process may be classified, based on their shape, into "normal operation" and "anomalous operation". Although such approaches have been widely publicised [Bulsari 1995], their statistical significance is limited by the fact that the number of examples of normal operation is usually (fortunately) much larger than the number of examples of faulty operation. Such an imbalance between classes is known to cause severe problems for the design of a classifier.

An alternative statistical approach consists in designing a statistical model of the normal operation of the process, and in detecting faults by comparing the prediction of the model of normal operation to the results of actual measurements made on the process; the existence of a statistically significant difference between the prediction of the model and the actual evolution of the process is a sign of anomalous operation.

Clearly, such an approach to fault detection requires the design of a very accurate model of the normal operation of the process, running in real time. Since the process under consideration is essentially non-linear, the use of neural networks is natural. However, it would be definitely wasteful to design a traditional "black-box" neural model, since a wealth of mathematical knowledge is available on the distillation process; therefore, we have developed a methodology for imbedding mathematical knowledge into the design of a dynamic neural model.

2.2 - A continuous-time model of the process

In a binary distillation process, mass and heat exchanges between a liquid phase and a vapour phase are organised in such a way that the vapour contains the most volatile product (the distillate) whereas the liquid contains phase contains the other product (the residue). At the top of the column, the distillate is condensed, partly collected for further processing and partly recycled into the process, whereas the residue is partly collected at the bottom of the column, and partly recycled. For better efficiency, the temperature gradient in the column is taken advantage of: the heat and mass exchanges occur at different levels of the column (the trays), so that the vapour phase is richer and richer in lighter components as it flows upwards, whereas the liquid phase is enriched in heavier components as it flows downwards [Lang et al. 1990, Rovaglio et al. 1990]. The raw material is fed into the column at an intermediate level (Figure 1).

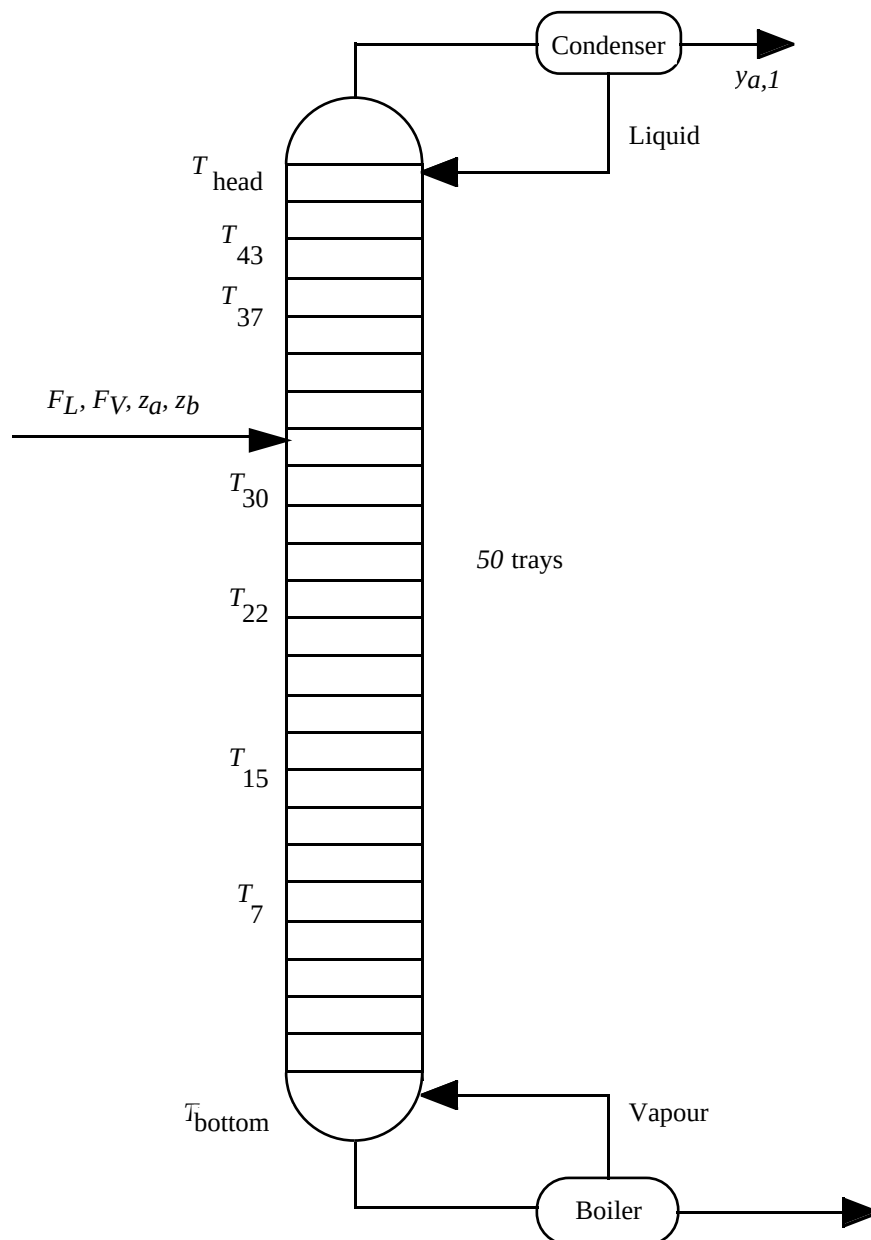


FIGURE 1
Schematic diagram of the distillation column.

We consider, as a first approach, that the column processes a mixture of three species a , b , c ; the relevant variables are the mole fractions x_a , x_b of components a and b in the liquid phase, and the mole fractions y_a , y_b of components a and b in the vapour phase; since mole fractions sum to one, the mole fractions of component c in the liquid and in the vapour are dependent variables. At each tray i , thermodynamic equilibrium at temperature T_i is assumed to be reached; we denote by $x_{a,i}$, $x_{b,i}$, $y_{a,i}$, $y_{b,i}$ the mole fractions at tray i . Since the mole fractions in the liquid phase are related to the mole fractions in the vapour phase by the equations of thermodynamics, there are only two independent variables per tray; in the following, we choose these to be $x_{a,i}$ and $x_{b,i}$, which will be state variables of the model. The equilibrium equations at tray i can be written as :

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

where P is the pressure in the column, and G_a and G_b are known non-linear functions. The mole fractions depend on temperature; since the measurements performed on the column are essentially the temperatures at various levels of the column, we are interested in the relation between the temperature and the state variables :

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

All the above quantities are temperature-dependent, since we are interested in a full dynamic model of the process. The differential equations which govern the process are the law of mass balance for components a and b :

$$M_i \dot{x}_{a,i} = L x_{a,i+1} + V y_{a,i+1} \pm L x_{a,i} \pm V y_{a,i}$$

$$M_i \dot{x}_{b,i} = L x_{b,i+1} + V y_{b,i+1} \pm L x_{b,i} \pm V y_{b,i}$$

where M_i is the number of moles retained at tray i , L is the liquid flow and V is the vapour flow. The exchanges occurring at tray i are shown schematically on Figure 2.

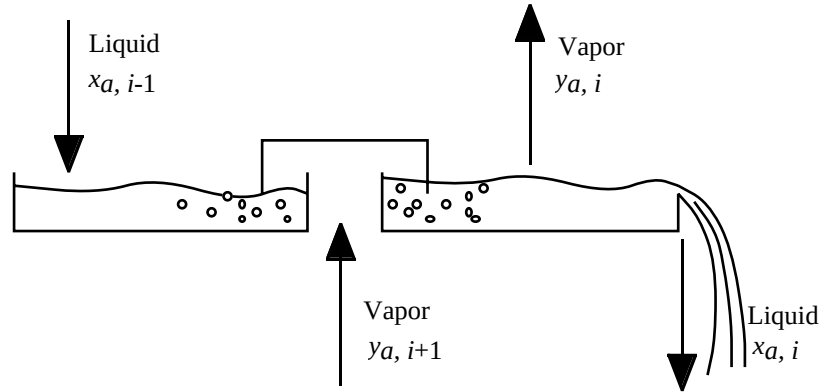


FIGURE 2
Heat and mass exchanges at tray i , which receives liquid from tray $i-1$ and vapour from tray $i+1$.

In addition, flow conservation must be taken into account, as shown on Figure 3.

2.3 - The distillation column

The distillation column whose modelling is reported here is an element of a steam-cracking unit of the French company Elf Atochem. It has 50 trays; the feeding of the column can be modelled as an additional, fictitious tray. Therefore, the model has 102 state variables. The measurements performed are : temperatures at 8 different locations of the column (see Figure 1), the mole fractions in the distillate, and the mole fractions in the residue. Thus, 2 state variables are measured directly, and 8 state variables are measured indirectly, through temperatures which, as mentioned above, are non-linear functions of the state variables. The input variables, which are measured, are P , L , V , F_L , F_V and the mole fractions of the components in the feed, z_a and z_b . Figure 4 is a schematic representation of the column.

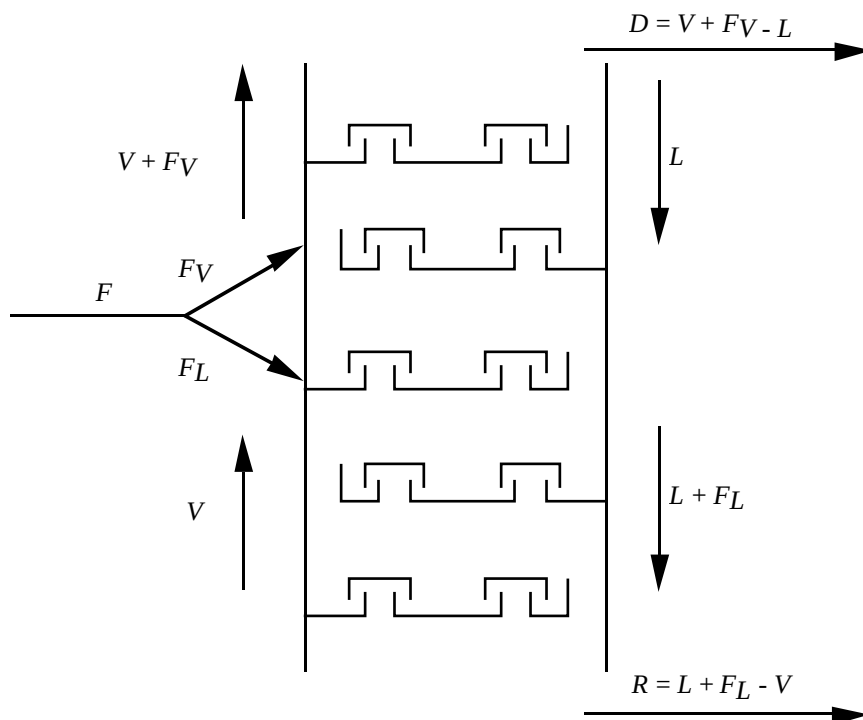


FIGURE 3

Liquid and vapour flows in a distillation column. F_L and F_V are the flows of liquid and vapour in the feed.

2.4 - The rationale of using a neural network for the simulation of the column

In order to simulate the above column, the natural procedure is to solve the equations of the model numerically. This, however, is not feasible, for several reasons. First, the model that has been presented is not accurate enough for the purpose of the application, i.e. early fault detection. An accurate description would require taking into account all the chemical compounds present in the column (instead of only three of them), and to measure accurately parameters which are specific to the column, and which are actually not measured. The number of equations thus involved precludes the resolution of the differential equations on a PC 486 in real time.

Therefore, it seems simpler to use a neural network model, which may be trained from sequences obtained through measurements made on the column. Training may be time-consuming, but, once the neural network has been trained, its operation is very fast.

Since the order of the model is known, it seems natural to use a state-space neural network, which reconstructs the full state of the column from the measurements; clearly, this is a hopeless task if the column is modelled as a dynamic *black-box* state-space neural model with 102 state variables: although the training of neural state-space models, from measurement of the outputs only, is known to be feasible, there is no guarantee whatsoever that the dynamics of the state variables of the model is identical to the dynamics of the state variables of the process. It has actually been shown on examples that a state-space neural network can model accurately the input-output relations of a simulated process, although state variables of the neural model are completely different from those of the simulated process [Rivals et al., 1995]. Conversely, if one makes use of the mathematical knowledge of the process in the design of the network, then state reconstruction is feasible, as will be shown below.

3 - FROM THE KNOWLEDGE-BASED MODEL TO THE DYNAMIC NEURAL MODEL

3.1 General presentation

A general presentation of the principles of knowledge-based neural modelling can be found in [Rivals et al., 1995]. In the present paper, we recall these basic principles with reference to the problem under consideration.

As shown above, an approximate model of the distillation column is available, in standard state-space form:

$$\frac{dx}{dt} = f[x(t), u(t)]$$

$$y(t) = g[x(t)]$$

where f and g are known analytically. Vector $x(t)$ has 102 components.

The state equations can be discretized to

$$x(k+1) = x(k) + f[x(k), u(k)] \quad (1)$$

$$y(k+1) = g[x(k+1)]$$

by Euler's method (other discretization techniques such as Runge-Kutta methods can be used as well). If two feedforward neural networks can be trained to approximate functions f and g , then a network such as shown on Figure 4 obeys the same discrete-time equations as the model.

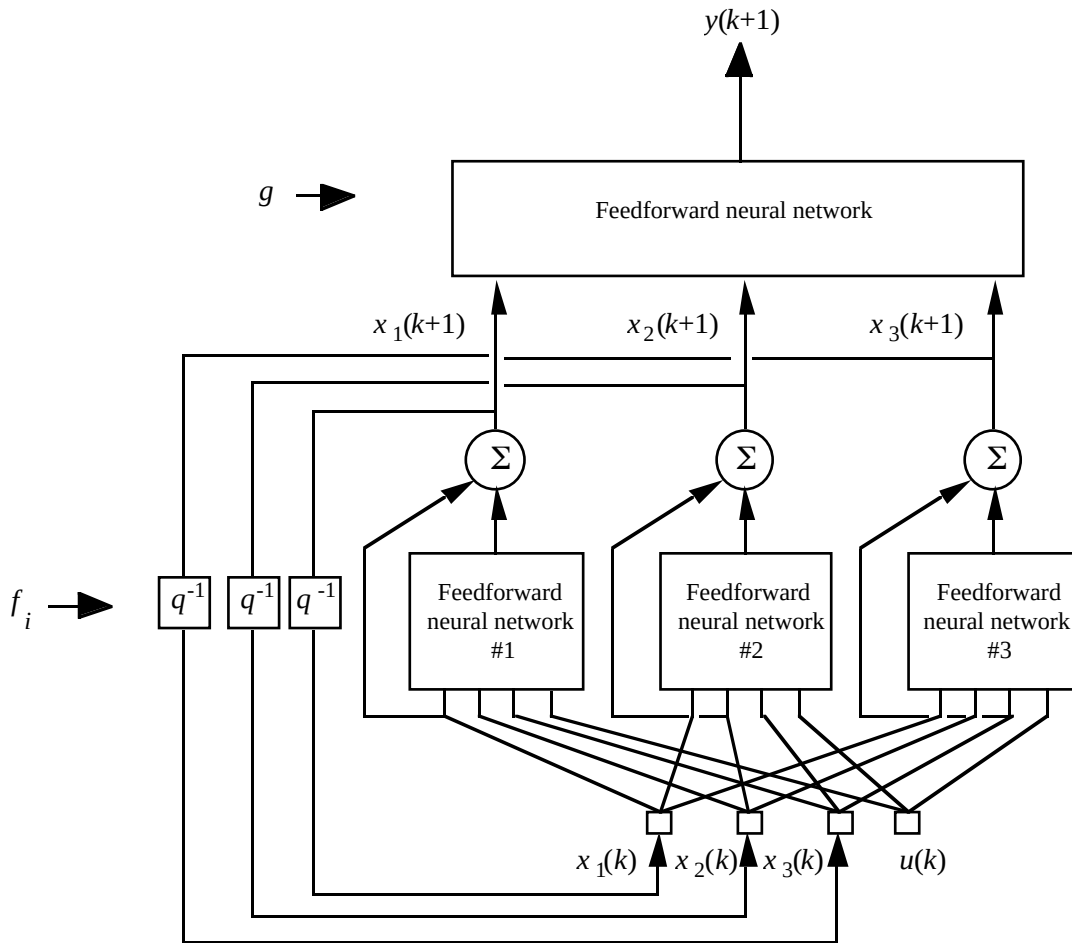


FIGURE 4
A third-order network obeying equations (1) (q^{-1} is the unit delay operator)

Since $x(k)$ is a vector, f is a vector too; therefore, instead of using a single network for approximating the whole vector f , it is generally advantageous to use different networks for different components f_i of f . In the application of this procedure to the distillation column simulator, each box shown on Figure 4 is a model of a tray. Similarly, function g of Figure 5 is the relation between the state variables and the measured temperature: it is function Θ as defined in section 2.2. This leads to a modular construction of the network, as illustrated on Figure 5.

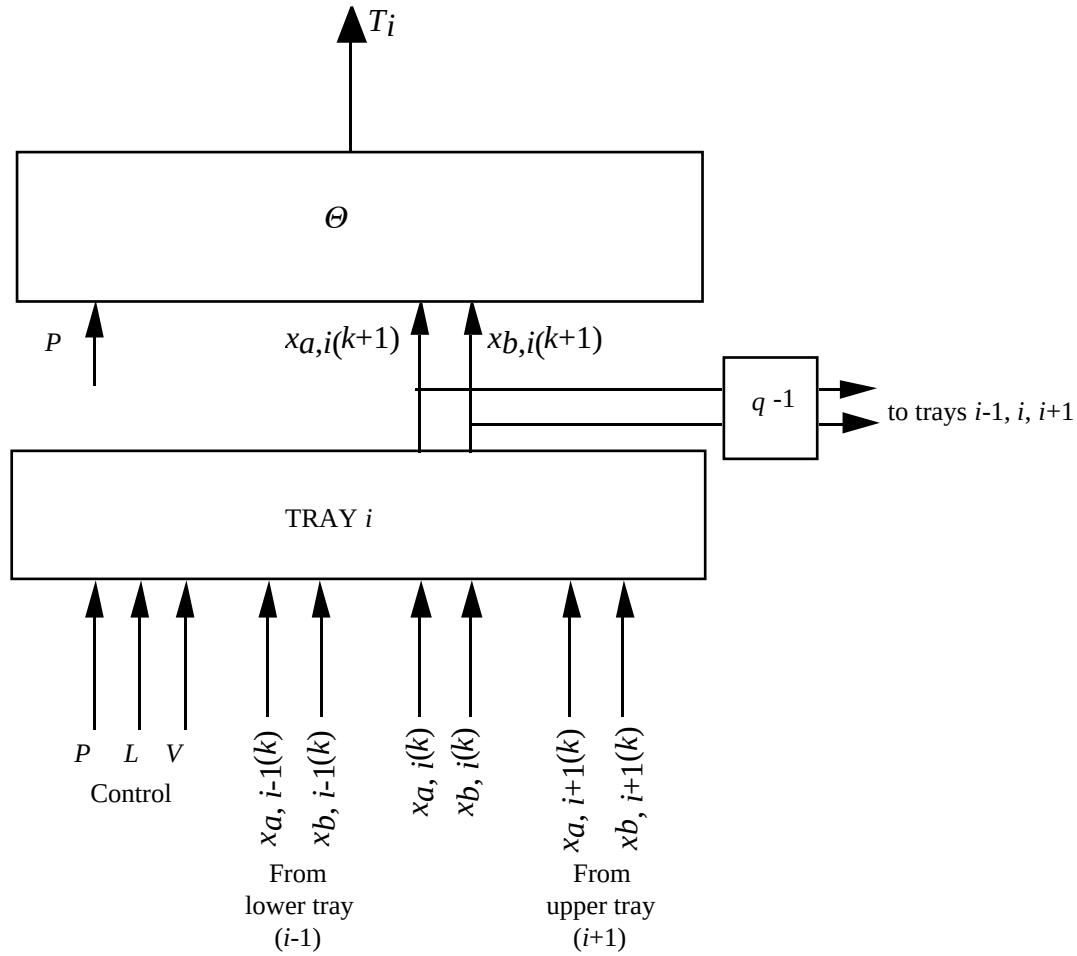


FIGURE 5

Modular construction of the dynamic neural network: neural model of tray i and of the relation between the state variables at tray i and the temperature T_i .

These networks can actually be trained from data generated by solving numerically the equations of a knowledge-based model such as the column model described above; this was performed using the PROSIM simulation package. The second-order algorithms used for training such recurrent neural networks are described in [Nerrand et al. 1993, Nerrand et al. 1994]

Thus, at the end of this step, one has a neural network which performs exactly as well, or as poorly, in terms of accuracy, as the state-space knowledge-based model, but which may require a much smaller computational effort than the numerical solution of the state equations.

As a final step, the knowledge-based neural model is trained with sequences measured on the process itself. In this step, not all weights are adjustable: since most weights of the network have a physical meaning, those which are known to be accurate and not to require any adjustment are kept fixed during training. The only adjustable weights are the weights of the black-box networks (if any), and the weights whose values are not known accurately from theory. Thus, one takes advantage of the intelligibility of a knowledge-based model, while retaining the flexibility and training capabilities of neural networks.

The final network obtained by applying this method to the distillation column model has 7 inputs, 10 outputs and 102 state variables. The number of neurons is slightly in excess of 1,000, but there are only 32 *adjustable weights*; this small number of weights results from the facts that (i) many weights are fixed since they have a physical meaning and are known accurately from thermodynamics, and (ii) extensive weight sharing occurs since all trays are identical. Therefore, this model is quite parsimonious in terms of adjustable weights.

4 - RESULTS

The model was trained and implemented on a standard PC 486 running at 66 MHz. One minute of computation allows the simulation of six minutes of real operation of the column, which is faster, by at least one order of magnitude, than standard dynamic models of equivalent accuracy.

The eight temperatures and two concentrations measured on the distillation column are sampled every three minutes. As mentioned above, a first training was performed on data generated by a knowledge-based simulator solving numerically the equations of the model; at the end of this step, the dynamics of the neural model was essentially correct, but it was not accurate enough for fault detection. A second training was performed from measured data, collected during one week of operation, containing typical normal operating modes and transients of the process, with a sampling period of 18 mn. Figure 6 shows a typical result concerning the week of training data; on this graph one point out of 6 is used for training; the other points are actually test data.

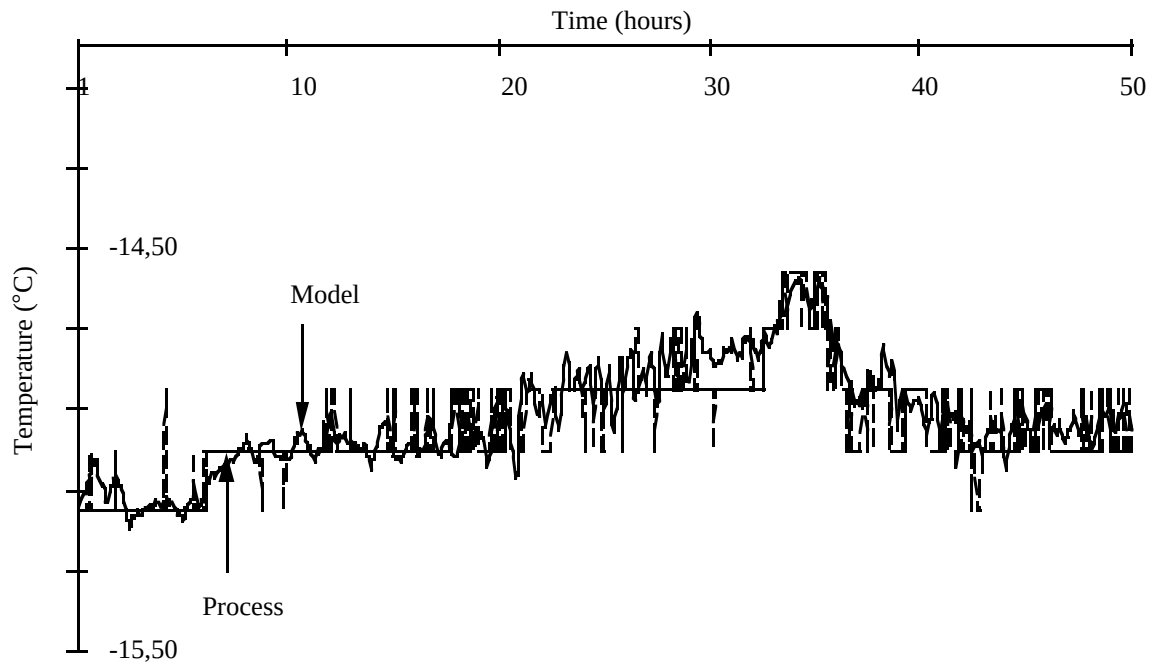


FIGURE 6

Typical results during the training period: temperature predicted and temperature measured at the top of the column. One point out of six is used as part of the training sequence; the other points are test points

The weights were subsequently kept fixed. Figure 7 shows the detection of a fault (excessive temperature) occurring one month after training: from $t = 0$ to $t = 70$ hours, the prediction of the model are in excellent agreement with the measurements performed on the process, despite rapid transients (around $t = 25$ hours for instance). Around $t = 72$ hours a sharp discrepancy appears, with a fast increase of the measured temperature ; this behaviour alone is not sufficient to conclude that an anomaly is occurring, since a similar fast increase occurs in normal operation (around $t = 25$ hours). After the corrective action taken by the operator (90 minutes after the inception of the fault), the temperature decreases and normal operation is resumed, as testified by the fact that the predictions of the model are in agreement with the measured temperatures. Since the predictions of the model were not available to the operator, the latter took the necessary corrective action 90 minutes after the inception of the fault; if the predictions of the model had been available to the operator, the discrepancy between the neural network prediction and the measurements would have been clearly apparent to him less than 20 minutes after the inception of the fault

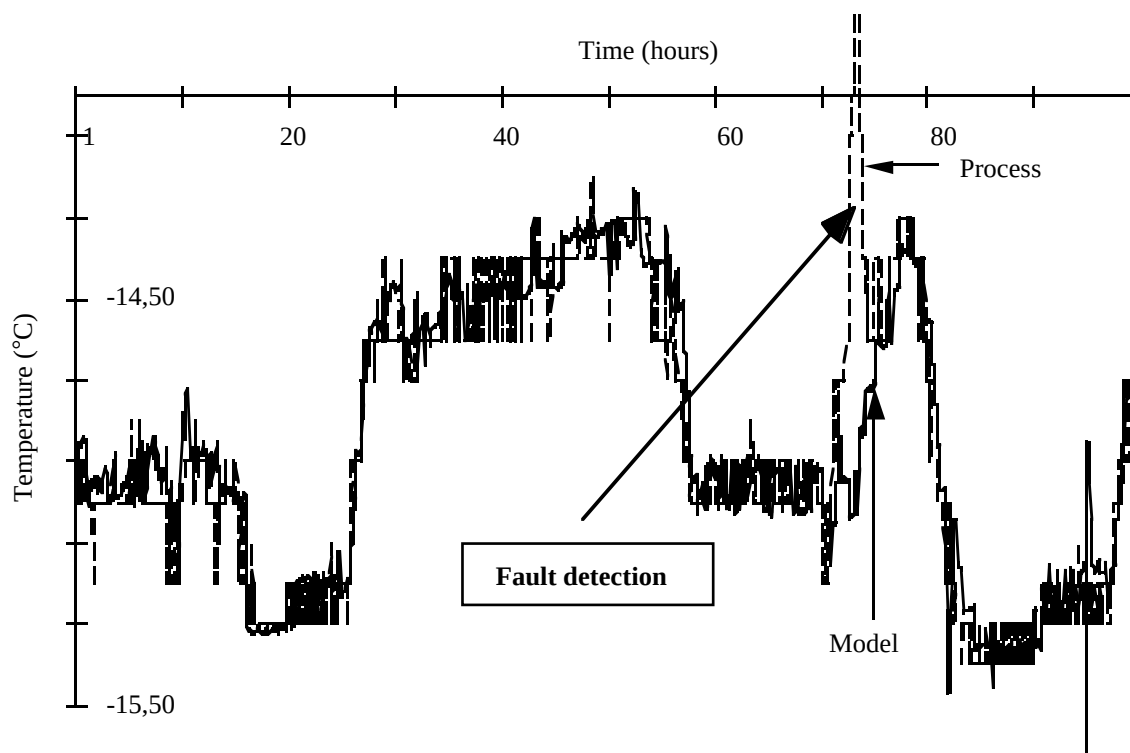


FIGURE 7
Detection of a fault.

CONCLUSION

It is the purpose of the present paper to show that dynamic neural networks are a powerful tool for accurately modelling complex, dynamic industrial processes. Provided a rigorous methodology is used, taking into account all mathematical information (albeit incomplete or inaccurate) available on the process, it is possible to design a network which actually reconstructs the state of the process from partial state measurements: far from being a traditional "black-box", such networks have the legibility of knowledge-based models, but have the flexibility of black-box models. Of course the application of knowledge-based dynamic neural modelling is not restricted to fault detection: such simulators can be used for computer-aided design, or for didactic purposes, or as elements of a global control loop.

ACKNOWLEDGEMENTS: the authors are very grateful to Jean-Pierre Corriou, who made this work possible.

References:

- Bulsari A.B. (1995) **Neural Networks for Chemical Engineers** (Elsevier Press, Amsterdam)
- 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.

Rivals I., Personnaz L., Dreyfus G., Ploix J.L. (1995) **Modélisation, classification et commande par réseaux de neurones: principes fondamentaux, méthodologie de conception et illustrations industrielles.** Les réseaux de neurones pour la modélisation et la commande de procédés, J.P. Corriou, ed. (Tec & Doc, Paris).

Rivals I., L. Personnaz, **Black-box Modeling with State-Space Neural Networks,** in : Neural Adaptive Control Technology I, R. Zbikowski and K.J. Hunt, eds. (World Scientific, 1995).

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.