Robust and reliable modelling for a distillation column

Yahya CHETOUANI

Université de Rouen, Département Génie Chimique, Rue Lavoisier, 76821 Mont Saint Aignan Cedex, France. Tel: (0033)235528466, Fax: (0033)235147191, Yahya.Chetouani@univ-rouen.fr

ABSTRACT

The main aim of this paper is to establish a reliable model of a process behavior both for the steady-state and unsteady-state regimes. The use of this accurate model allows distinguishing a normal mode from an abnormal one. Therefore the neural black-box identification by means of a NARX (Nonlinear Auto-Regressive with eXogenous) model has been chosen. It shows the choice and the performance of the neural network. The model is implemented by training a Multi-Layer Perceptron Artificial Neural Network (MLP-ANN) with input-output experimental data. After describing the system architecture, a realistic and complex application as a distillation column is presented in order to illustrate the neural model reliability.

1. INTRODUCTION

In the last few years ever-growing interest has been shown in production quality standards and pollution phenomena in industrial environments. However process development and continuous request for productivity led to an increasing complexity of industrial units. The dynamic nature and the nonlinear behavior of such units pose challenging control system design when products of constant purity are to be recovered.

In chemical process control, the processes typically exhibit nonlinear behavior. The intrinsic highly nonlinear behavior in the industrial process, especially when a chemical reaction is used, poses a major problem for the formulation of good predictions and the design of reliable control systems [1, 2]. Similar problems arise also from the uncertainty for the parameters of the process, such as the reaction rate, activation energy, reaction enthalpy, heat transfer coefficient, and their unpredictable variations.

Most large-scale process models derived from first principles are represented by nonlinear differential–algebraic equation (DAE) systems. Since such models are often computationally too expensive for real-time control. Even if global nonlinear models are available, they may not be appropriate for control and monitoring or fault detection (FD) purpose [3].

On the other hand, when fundamental process knowledge is unavailable or incomplete, or when the resulting model may not be particularly suitable for FD applications, the input/output models identified from plant data may be more useful. Nonlinear autoregressive models have been used [6], as well as models that use a set of polynomial basis functions (e.g. Volterra functions). Also ANNs provide an excellent
mathematical tool for dealing with severe nonlinear problems. Another attractive property is self-learning ability. As a result, nonlinear systems can be modeled with a great flexibility. These features allow one to employ artificial neural networks to model complex, unknown and nonlinear dynamic processes.

Distillation is one of the most important separation processes used in many chemical industries. It has extensively been studied [4, 5]. There are essentially two approaches by which nonlinear models can be developed for a distillation column; from first principles by using the process knowledge or empirically from input/output data. The advantages and disadvantages of each approach are well known. In industrial practice, it is not always possible in general to obtain accurate first principles models for high-purity distillation columns.

Most industrial columns are used to separate multi-component mixtures whose constituent elements are often not known completely; the fundamental thermodynamics of multi-component vapor-liquid equilibrium, the physical property data, and other essential constitutive relations required for the successful development of a first principles model are not always available. And even when such knowledge is available, the resulting models usually occur in the form of a very large system of coupled nonlinear ordinary differential equations, and may therefore not always be the most convenient for controller or fault detection (FD) design.

The purpose of this study is to obtain a reduced and reliable model which allows reproducing the dynamics of a nonlinear process as a distillation column. This reliable model enables to reproduce the process dynamics under all operating conditions i.e. steady-state or unsteady state. The present study focuses on the development, and implementation of a NARX neural model for the forecasting of the distillation column dynamics. Experiments were performed in a distillation column and experimental data were used both to define and to validate the model. The performance of this neural model was then evaluated using the performance criteria. Results show that the NARX neural model is representative for the dynamic behavior of this nonlinear process. The experimental set-up, modelling procedure, and prediction results are described in the following sections.

2. EXPERIMENTAL SET-UP

The feed tank (fig. 1) contains a mixture to be separated (Toluene-Methylcyclohexane) with a mass composition at 23 % methylcyclohexane. The operation in continuous mode involves charging the still with the mixture to be separated, bringing the column to equilibrium under total reflux. The product is introduced through the optimal feed tray so that the light components are volatilized, while the heavy part goes down again with the reflux in the column reboiler. The quality of the collected top product of the column depends on the reflux flow rate. The reflux ratio is varied through the magnetic valve by changing the relative quantities of material returning to the column and flowing to product storage. Feed preheating system is constituted by three elements of 250 W each one. In addition it has a low liquid level switch in order to avoid the
running if the level is excessively low. The reciprocating feed pump is constituted by a membrane allowing firstly the suction of the mixture and the discharge towards the tank with a flow capacity \( F = 4.32 \text{ L.h}^{-1} \). The column has also a reboiler of 2 liters hold-up capacity, an immersion heater of a power \( Q_b = 3.3 \text{ kW} \) and of a level liquid switch sensor which allows the automatic stop of heating if the level is insufficient. The stirring of the mixture in the reboiler is ensured by the boiling mixture. The internal packing is made of Multiknit stainless 316L which enhances the mass transfer between the vapor and liquid phases. In order to approach the adiabatic conditions, heat-insulation made of glass wool is laid around the column. A condenser is placed at the column overhead in order to condense the entire vapor coming out from the column. The cooling medium used in exchangers is water. The heat-transfer area of the total overhead condenser is 0.08 m². Moreover the reflux timer \( (R_t) \) allows to control the reflux ratio \( (R_r) \). It is monitored by the overhead product temperature \( (T_d) \). When the required distillate temperature \( (T_d) \) is attained, the reflux timer opens. In the opposite case, it remains closed. Distillation supervision control system allows to modify the parameters and to follow their evolution such as the pressure drop \( (\Delta P) \), the flow or the temperatures at different points of the distillation column. This control system, therefore, must hold product compositions as near the set points as possible. The thermocouples are coupled to a calibrated amplification circuit (4-20 mA, 0–150°C) whose signals are inputted to the computer online, which permits the bottom and top temperatures to be obtained. The unit has twelve sensors which measure continuously the temperature throughout the column.

![Figure 1. Experimental device: Distillation column](image)

3. IDENTIFICATION OF NARX MODELS

The NARX neural model used to describe accurately the process behavior is the classical MLP-ANNs [7, 8] with one layer of hidden neurons. The NARX model of a finite dimensional system [9] with order \( (n_y, n_u) \) and scalar variables \( y \) and \( u \) is defined by:

\[
y(t) = \phi(y(t-1), \ldots, y(t-n_y), u_i(t-n_k), \ldots, u_i(t-n_k-n_u)) \quad 1 \leq i \leq m
\]  

(1)
where \( y(k) \) is the Auto-Regressive (AR) variable or system output; \( u(k) \) is the eXogenous (X) variable or system input. \( n_y \) and \( n_u \) are the AR and X orders, respectively. \( m \) is the number of the used inputs. \( n_k \) is the time delay between \( u \) and \( y \). \( \phi \) is a nonlinear function.

The feed-forward MLP-ANN is used in this study and consists in a large number of highly connected nonlinear simple neurons. The log-sigmoid transfer function is used on the hidden layer and a linear transfer function on the output layer. A back-propagation training function for feed-forward networks using momentum and adaptive learning rate techniques is used.

### 3.1. Calculation of the NN output

Though the applicability of neural networks to solve several nonlinear complex problems has been amply demonstrated, the time taken to train neural networks off-line can be quite excessive [10, 11]. The following steps explain the calculation of the ANNs output based on the input vector.

1. Assign \( \hat{w}^T(k) \) to the input vector \( x^T(k) \) and apply it to the input units where \( \hat{w}^T(k) \) is the regression vector given by the following equation:
   \[
   \hat{w}^T(t) = [y(t-1), ..., y(t-n_y), u(t-1-n_k), ..., u(t-n_u-n_k)]
   \]  
2. Calculate the input to the hidden layer units:
   \[
   net^h_j(k) = \sum_{i=1}^{p} W_{ji}^h x_i(k) + b^h_j
   \]  
   where \( p \) is the number of input nodes of the network, i.e. \( p = n_y + n_u + n_h \); \( j \) is the \( j \)th hidden unit; \( W_{ji}^h \) is the connection weight between \( i \)th input unit and \( j \)th hidden unit; \( b^h_j \) is the bias term of the \( j \)th hidden unit.
3. Calculate the output from a node in the hidden layer:
   \[
   z_j = f^h_j (net^h_j(k))
   \]  
   where \( f^h_j \) is the log-sigmoid transfer function.
4. Calculate the input to the output nodes:
   \[
   net^q_l(k) = \sum_{j=1}^{h} W_{jl}^q z_j(k)
   \]  
   where \( l \) is the \( l \)th output unit; \( W_{jl}^q \) is the connection weight between \( j \)th hidden unit and \( l \)th output unit.
5. Calculate the outputs from the output nodes:
   \[
   \hat{v}_l(k) = f^q_l (net^q_l(k))
   \]  
   where \( f^q_l \) is the linear activation function defined by:
   \[
   f^q_l (net^q_l(k)) = net^q_l(k)
   \]
3.2. Back-propagation training algorithm

The error function $E$ is defined as:

$$E = \frac{1}{2} \sum_{l=1}^{q} (v_l(k) - \hat{v}_l(k))^2$$  \hspace{1cm} (8)

where $q$ is the number of output units and $v_l(k)$ is the $l$th element of the output vector of the network. Within each time interval from $k$ to $k+1$, the back-propagation (BP) algorithm tries to minimize the error for the output value as defined by $E$ by adjusting the weights of the network connections, i.e. $W_{ji}^h$ and $W_{lj}^q$. The BP algorithm uses the following procedure (Eqs.9, 10, 11, 12):

$$W_{ji}^h(k+1) = W_{ji}^h(k) + \alpha \Delta W_{ji}^h(k) - \eta \frac{\partial E}{\partial W_{ji}^h(k)}$$  \hspace{1cm} (9)

$$W_{lj}^q(k+1) = W_{lj}^q(k) + \alpha \Delta W_{lj}^q(k) - \eta \frac{\partial E}{\partial W_{lj}^q(k)}$$  \hspace{1cm} (10)

where $\eta$ and $\alpha$ are the learning rate and the momentum factor, respectively; $\Delta W_{ji}^h$ and $\Delta W_{lj}^q$ are the amounts of the previous weight changes; $\partial E/\partial W_{ji}^h(k)$ and $\partial E/\partial W_{lj}^q(k)$ are given by:

$$\frac{\partial E}{\partial W_{ji}^h(k)} = -[z_j(k)(1-z_j(k))x_i(k)] \sum_{l=1}^{q} [(v_l(k) - \hat{v}_l(k))\hat{v}_l(k)W_{lj}^q(k)]$$  \hspace{1cm} (11)

$$\frac{\partial E}{\partial W_{lj}^q(k)} = -(v_l(k) - \hat{v}_l(k))z_j(k)$$  \hspace{1cm} (12)

The implementation of the ANN for forecasting is as follows:

1. Initialize the weights using small random values and set the learning rate and momentum factor for the ANN.
2. Apply the input vector given by Eq. 2 to the input units.
3. Calculate the forecast value of the error using the data available at $(k-1)$th sample (Eqs.2, 3 4, 5, 6, 7).
4. Calculate the error between the forecast value and the measured value.
5. Propagate the error backwards to update the weights (Eqs. 9, 10, 11, 12).
6. Go back to step 2.

For weights initialization, the Nguyen-Widrow initialization method [12] is best suited for the use with the sigmoid/linear network which is often used for function approximation.
4. RESULTS AND DISCUSSION

In order to select the optimal number of hidden neurons, tests were performed by varying the number of neurons between 5 and 25 for the temperature ($T_d$) modelling. The minimal number of inputs is avoided to ensure the model flexibility. Also, the maximum number of inputs is excluded to avoid the over-fitting.

$$T_d(t) = \phi(T_d(t-1), R_i(t-2), \Delta P(t-3), Q_f(t-4), T_j(t-6), F(t-8), T_d(t-1) - \hat{T}_d(t-1))$$

The evolution of the loss function is given in fig. 2 in the training and the validation database. This figure shows that the neurons number in the hidden layer in training can be 7, 10 or 21. However this number can slightly change and becomes 8, 10 or 14 in the test.

The models which have a structure (7-8-1 and 7-10-1) exhibit the acceptable LF. Based on engineering judgment, the model 7-8-1 would be preferred without significant loss of accuracy. Fig. 3 shows the difference between the experimental overhead product temperature and those simulated by the neural model.

**Figure 2. Loss functions**

**Figure 3. Prediction error of the temperature ($T_d$)**
After analyzing this figure, it emerges that the NARX model 7-8-1 for modelling the temperatures \( (T_d) \) ensures satisfactory performances as it is indeed able to correctly identify the dynamics of the distillation column. Therefore, the reduced neural model is considered as reliable one for describing the dynamic behavior of this column. There is a good agreement between the learned neural model and the experiment in the validation phase.

In conclusion the main advantage of the neural approach consists in the natural ability of neural networks in modelling nonlinear dynamics in a fast and simple way and in the possibility to address the process to be modeled as an input-output black-box, with little or no mathematical information on the system.

5. CONCLUSION

This work aims to identify process dynamics by means of a NARX model. The identification of the system dynamics by means of input-output experimental measurements provides a useful solution for the formulation of a reliable model. In this case, the results showed that the model is able to give satisfactory descriptions of the experimental data.

The developed neural models are used in a recursive scheme in order to test their ability to perform the behavior prediction. Finally, the identified neural models will be useful for the detection and the isolation (FDI) of faults which can occur through the process dynamics.

6. REFERENCES