COMPOSITION INFERENCE USING NEURAL NETWORKS AND INFERENTIAL CONTROL IN THE ETHYLBENZENE PROCESS

Emanuella Francisca de Lacerda Vieira

Paulo Romero de Araujo Mariz

Marcelo da Silva Pedro

Jonas Laedson Marinho da Silva Santos

Leopoldo Oswaldo Alcazar Rojas

Arioston Araujo de Morais Junior

emanuella_vieira@hotmail.com

rom3romariz@gmail.com

marcelopedrocz@hotmail.com

jonas.laedson@hotmail.com

leopoldo@ct.ufpb.br

aamj@ct.ufpb.br

Federal University of Paraiba

Cidade Universitária, s/n - Castelo Branco III, João Pessoa - PB, 58051-900. Brazil

Abstract. *For the implementation of process control systems focused on optimization, such as advanced control, it is necessary to continuously measure the properties of various plant streams, like the product composition of a process unit. However, in most cases, there are no real-time measurements of these quantities, or these data are only available at a very low frequency. Therefore, soft sensors are increasingly applicable in this area, and they consist of algorithms that can be estimate, from measurable variables at high frequencies (such as flow rates, pressures, temperatures, other compositions available), composition of products whose direct measurement is not possible or can't be performed at the desired frequency. The use of Artificial Neural Networks (ANNs) for the construction of soft sensors has been gaining more and more space, due to their non-linearity, generalization and adaptability of complex problems. This work aims to construct a methodology of soft sensors for a productive process of ethylbenzene (EB), which consists of two reactors in series and two columns of distillation with liquid recycle streams, and was interesting for the study due to occurrence of transient effects in the high purity compositions of the top and bottom streams of the second process distillation column. A methodology was built in stages, being the first stage of modeling,*

Proceedings of the XXXVIII Iberian Latin-American Congress on Computational Methods in Engineering P.O. Faria, R.H. Lopez, L.F.F. Miguel, W.J.S. Gomes, M. Noronha (Editores), ABMEC, Florianópolis, SC, Brazil, November 5-8, 2017.

followed by simulation (software Aspen Plus and Dynamics^{TM}) and equipments sizing involved in the process. Then, a key-variable selection algorithm with multivariate statistics was developed with the methods of all possible regressions and stepwise regression. Thereafter, were used two types of ANN architectures: multi-layer perceptron and Elman's recurrent network, to estimate and control compositions using Matlab® software. The results obtained showed that the ANNs were able to approximate the values of the compositions of the real values obtained by the mathematical model, and the estimates were evaluated by means of the mean square error (MSE) and root mean square error (RMSE). The measurement provided with the soft sensor was used to implement inferential feedback control, thereby was possible to minimize the transient effects of characteristic disturbances inserted in the process.

Keywords: *Artificial neural network, ethylbenzene, soft sensor, inferential control*

INTRODUCTION

Industries modify and improve their technologies when they are required to increase their productivity, effective safety conditions, efficiency indices and final quality of the product. In this context, the importance of monitoring critical process variables using appropriate measurement devices is evident. For real-time monitoring of physical variables of chemical processes industries are used numerous physical sensors. However, when it is desired to measure, in real time, non-easily measurable chemical or biochemical variables, such as the composition of the products of distillation columns, there are many obstacles to monitoring: no measurement, high cost or no measurement in time (FORTYA et al., 2007; MORAIS JR, 2015). Thus, this work proposes a methodology for the construction of virtual sensors using inference models for the measurements of composition through artificial neural networks (ANN) as a viable alternative to overcome these and other presented difficulties.

Some models of systems identification are used for inference of chemical variables, among which artificial neural networks stand out. The solution of problems through ANN is quite attractive, given its main characteristic: the parallelism. This feature resembles the way the brain handles information received by its neurons and thus creates the possibility of superior performance in solving problems based on conventional linear and phenomenological models.

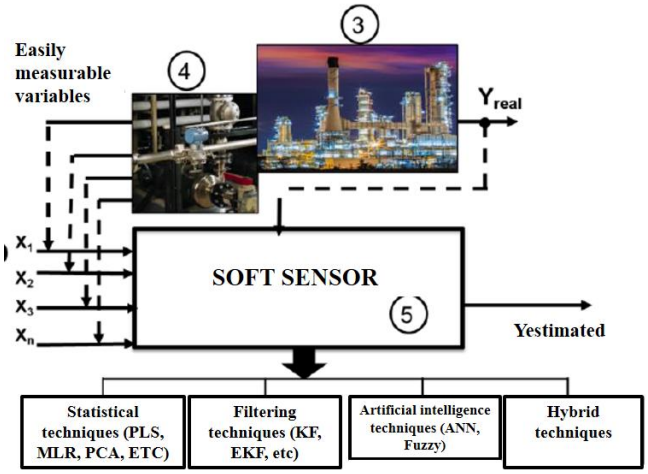
Initially, the knowledge of the process (engineering sense) can't and shouldn't be discarded, but by itself is not enough to select some important secondary input variables for the construction of virtual sensors. In this case, variable selection techniques, such as multivariate statistical analysis (exhaustive and sequential search) and methods based on artificial intelligence (for example, the genetic algorithm), are commonly used.

The object of study of this work is a chemical plant producing ethylbenzene (EB), an aromatic chemical compound in the liquid state at room temperature and highly flammable, presents itself as a great chemical commodity. Its production represents the beginning of the production chain of styrene, a monomer with a large market, with a demand of more than 25 million tons per year worldwide. In Brazil, the consumption is of 600 thousand tons of styrene per year, being a good part transformed into polystyrene, with about 32% absorbed by the disposable industries (cups), 22% with the industries of white line, 16% destined to the packaging of frozen food, and the other 30% are with the electronics and automotive industries. On a smaller scale, it is used by the paint and polyester resin industries. The

production process used in this work consists of two series reactors two distillation columns and two liquid recycle streams (LUYBEN, 2011).

1 SOFT SENSOR

Soft sensors are mathematical algorithms implemented in software, used mainly for inference of variables not easily measurable, from secondary variables of easy measurement, such as physical measurements. This alternative arises from an operational difficulty or from the high cost of obtaining the desired variable, for example estimating compositions of distillation product streams. The configuration of a virtual sensor implemented in software is presented according to Figure 1:



Font: adapted of MORAIS JR, 2015

Figure 1. Soft sensor implemented in software for variable inference

The fact that human beings are able to conduct complex tasks under significant uncertainty has stimulated research by alternative modeling patterns, such as Intelligent Systems (IS) or Artificial Intelligence (AI) techniques. IS is methodologies that employ techniques motivated by biological systems and human intelligence in the development of dynamic systems models. In this way, they are used in virtual sensor construction models, among which artificial neural networks (ANN) stand out.

2 ARTIFICIAL NEURAL NETWORKS (ANN)

Artificial neural networks are processors equipped with simple processing units that store knowledge acquired experimentally through learning and make it available for use (HAYKIN, 2001). These two characteristics make them computational models that resemble structure and functioning to the nervous system of living beings.

ANN's require low computational load and can provide an accurate representation of the behavior of the process due to its learning ability and generalization. Thus, the network is able to produce suitable outputs, even for inputs that were not present in the training process.

Other important characteristics are the capacity for self-organization and temporal processing, in addition to non-linearity, which, together with those previously mentioned, make them an extremely powerful and attractive computational tool for solving complex problems2.1

2.1 Structure of an artificial neuron

A neuron is the basic information-processing unit for the operation of a neural network, which in turn consists of a structure formed by the interconnection between these process elements (HAYKIN, 2001).

The general model for an artificial neuron is shown in Figure 2 below:

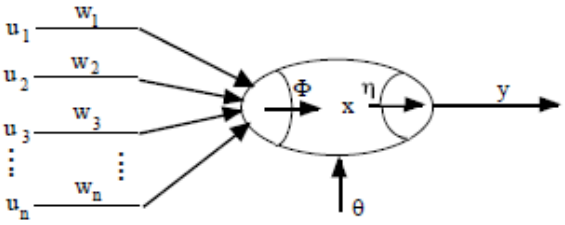


Figure 2. Artificial neuron

In this model, inputs named u1, u2, ..., un are weighted by their synaptic weights, which are the connecting forces between the neurons. The function Φ linearly combines all input signals weighted by their respective synaptic weights, according to Equation 1, to produce an activation potential value. The activation function η, which produces the output of the neuron, limits it to a range of reasonable values to be assumed. An auxiliary value θ, known as bias, is generally used to increase or decrease the net input of the activation function.

$$\Phi = \sum_{i=1}^{n} u_n w_n \tag{1}$$

The activation functions can be of two types: fully differentiable, whose first-order derivatives exist throughout the domain of the function, or partially differentiable, which have points whose derivatives are non-existent. The main representatives are used in Table 1 below, and their graphs in Figure 3:

Table 1. Examples of activation functions.

Fully differentiable functions	
Linear function	$f(\Phi) = a\Phi$
Hyperbolic tangent function	$f(\Phi) = \frac{1-e^{-\alpha\Phi}}{1+e^{-\alpha 2\Phi}}$
Logistic function	$f(\Phi) = \frac{L}{1+e^{-k(\Phi-\Phi 0)}}$

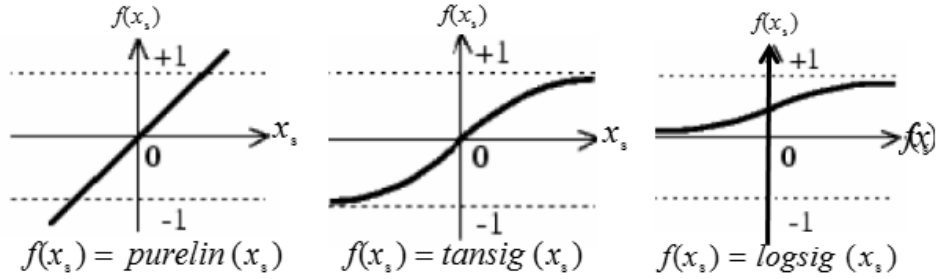


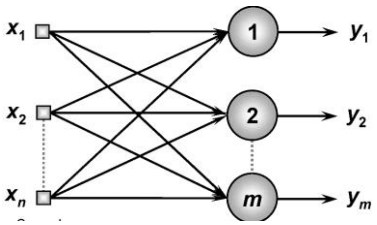
Figure 3. Main activation functions graphics.

2.2 Network Architectures

The way in which the neurons of a network are arranged is called network architecture (HAYKIN, 2001). For its definition, parameters such as number of network layers, number of nodes in each layer, type of connection between nodes and network topology are required. It is worth mentioning that the structure of an ANN is composed of the input layer, where there is reception of data, measurements or patterns presented to the network; By hidden layers, responsible for data processing and extraction of the characteristics of the process; And the output layer, which presents the final result of the network.

Three types of architectures are commonly used:

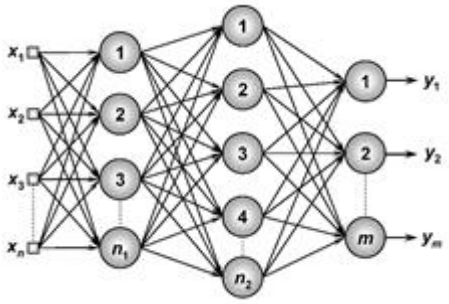
- *Single layer feedforward networks:* there are no additional layers between the input layer and the output layer (Figure 4). The input layer is not considered in this count since its role is to distribute the entrances of the networks to the next layers (MORAIS JR, 2015). The propagation of signals is done in a unidirectional way: the pulse follows from the input towards the single layer of the network, where the final result is processed and presented. This model is generally employed in pattern classification and filtering problems.



Font: SILVA et al., 2010

Figure 4. Single layer network representation

- *Multi-layered feedfoward networks*: considered the most used in solutions of engineering problems (Figure 5), stand out in solving some classes of problems, such as: pattern recognition, function approximation, dynamic systems, Identification and control of processes and optimization of systems (SILVA, 2010). In this configuration, the output of each neuron from a preceding layer is input to all neurons in the next layer. The addition of one or more hidden layers makes the network capable of extracting high order statistics (HAYKIN, 2001).

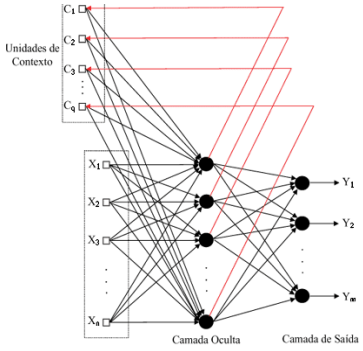


Font: SILVA et al., 2010

Figure 5. Multi-layered feedforward networks representation

The signal is received by the input layer and sent to the neurons of the first hidden neural layer, which in turn sends the signals as input to the second layer, and so on, on a unidirectional pulse until it reaches the output layer. The intermediate layers extract most of the information and process it through the synaptic and bias weights of their neurons, thus adequately representing the environment of the system in question. The neurons in the output layer receive the stimuli from the last hidden layer and provide a response pattern as output from the network.

• *Recurrent networks:* they are neural networks with global feedback dynamics, that is, they present "cycles" in their connections, that is, the output of neurons of a layer i are inputs of neurons of an i-j layer, with j> = 0. There are feedback loops between the network output signals to those of the input layer or any other layers hidden along the architecture, as shown in Figure 6. The use of the recurrent structure is a way of including temporal aspects in the operation of the neural network, which adds dynamic characteristics, that is, makes the network more sensitive to signals that vary in time. A recurrent network needs to have temporal memory (ELMAN, 1990).



Font: SEGATTO & COURY, 2006.

Figure 6. Elman Neural Network

2.3 Learning and training an ANN

The ability to learn associated with a neural network is one of the most striking features and makes it similar to the nervous system of living beings. It is the ability to adapt to existing rules or to the environment, making its performance variable over time.

Learning process consists of the following sequence of steps (HAYKIN, 2001): stimulus of the environment, consequent modifications in the free parameters of the network and new response to the environment with the modifications suffered in its internal structure.

Learning occurs through network training, an iterative process of modifying and adjusting the synaptic weights through a set of well-defined rules. There are two types of methods for training ANNs: supervised and unsupervised training.

In supervised training, knowledge is represented by a set of input-output examples. It is like a teacher-student situation: the teacher can show the network a desired response to a training vector. The parameters are then adjusted according to the training and the error signal, which is the difference between the expected response and the actual network response. When the error is minimal, the training can be terminated. Already in the unsupervised, there are no examples of the function to be taught to the network, nor is there a teacher to oversee the learning process; The parameters are optimized in relation to a measure independent of the task.

2.4 The Levenberg-Maquardt training algorithm

The backpropagation algorithm has a slow convergence process, so alternative methods have been developed to obtain a higher convergence speed. The Levenberg-Marquardt algorithm (LMA) represents the bridge between Newton's method and Gradient descent. Given a starting point x_0, a series of vectors x_1, x2, ..., xn is produced until it converges to a local minimum x *, for which the function must be adjusted. Such algorithm needs to store a square matrix with dimension in the order of the number of connections, so it is important to use a network with fewer than a hundred connections.

Consider F (w) which is a second order function in terms of the weight vector w, g its vector gradient and H its Hessian matrix. According to the LMA, the ideal fit Δw applied to the parameter vector w through Equation 2:

$$\Delta \mathrm{w} = \mathrm{w}_{\mathrm{K}+1} - \mathrm{w}_{\mathrm{K}} = \left[\mathrm{H} + \mu\, \mathrm{I}\right]^{-1} g \tag{2}$$

Where ***I*** is identity matrix of same dimensions as **H** and **μ** is a *regularizing*, or *loading, parameter* that forces the sum matrix (H+**μ**I) to be positive definite and safely well conditioned throughout the computation.

Hessian's complexity nature H (w) represents a computational limitation, its best representative form and with a smaller computational effort can be expressed through Equation 3, the desired adjustment being calculated for each iteration of the LMA with the cost function ξ_{fc} (w).

$$H(w) \approx \frac{\partial^2 \xi_{fC}(w)}{\partial w^2} = \frac{1}{N} \sum_{i=1}^{N} \left[\frac{\partial F(x(i); w)}{\partial w} \right] \left[\frac{\partial F(x(i); w)}{\partial w} \right]^T \tag{3}$$

The use of this approximation is justified when the LMA is operating in the vicinity of a local or global minimum (LEVENBERG, 1994). Thus, LMA is a first-order optimization

method that is suitable for non-linear least squares estimation problems. In addition, due to the fact that both equations involve the mean over the training sample, the algorithm is a batch form.

3. VARIABLE SELECTION

In order to construct an inference model, it is important to select an adequate set of variables that probably includes all the most important ones, that is, it is necessary to filter which of the process inputs most influence the desired response variable. For this, we use techniques of selection of variables, search techniques that aim to find the best regression equation, that is, the best subset of variables when there is a large number of candidate variables.

The selection of variables that will be part of the model is performed by comparing the models derived from the combinations of the available variables. The comparison should be made systematically, and thus, evaluation criteria are available to evaluate, compare and quantify the fit of the model to the available experimental data.

Among the existing methodologies, the exhaustive search methods and the sequential or stepwise methods stand out. The exhaustive search tests all possible combinations for a set of candidate entries, generating a template for each subset of variables, which is evaluated through the criteria quoted above. Sequential methods, however, evaluate a smaller subset of variables (Montogmery and Runger, 2014) by evaluating the effect of addition or sequential withdrawal of variables.

3.1 All Possible Regressions

Comprehensive search method that requires adjustment of all regression equations involving 1, 2, ..., n candidate variables. Consider a multiple linear regression model and its assumptions, Equation 4:

$$\hat{y} = \beta_o + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p \tag{4}$$

The methodology evaluates all sub models composed of the possible subsets of the variables p and identifies the best of these subsets through the evaluation criteria, mentioned later. It should be noted that there must be a constant term in any subset available for the model. Thus, if there are K regressors, 2^K regression equations will be generated containing the variables, and, as the candidates increase, the number of equations increases rapidly: doubles with each addition of a new candidate variable. Therefore, it isn't recommended when the number of regressors is reasonably large (Montgomery and Runger, 2014).

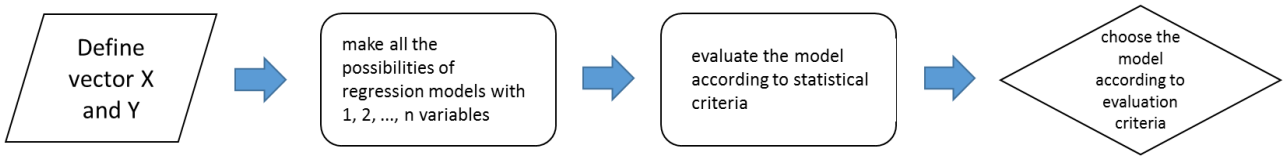


Figure 7. All possible regressions algorithm steps

3.2 Stepwise Regression

It is a technique widely used in variable selection, whose iterative procedure develops a sequence of models by adding or removing regressors candidates at each step, taking into account the influence of the input or output of each variable in the model. The most commonly used criterion to evaluate the need to add or remove regressors in each step is the partial F test, according to Equation 5:

$$F_j = \frac{SQR\ (\beta_2,\beta_0)}{MQE\ (x_1,x_2)} \tag{5}$$

SQR is the quadratic sum of the residuals and MQE is the quadratic mean of the error.

It starts with the model containing a variable, which is the one with the highest correlation with the output variable Y. In the next step, the remaining candidate regressors are analyzed, and the variable for which the partial statistic F is maximum is added to the model, if the calculated F value is greater than a previously set value. It is also analyzed if there is a need to remove the variable added in the previous step by calculating the F. If the result of F is less than a pre-set value fsai, the previously added variable is removed and the model proceeds to the next step using the same dynamics. The procedure for when no regressor can be added or removed.

3.3 Statistical Evaluation Criteria

The selection procedure can be evaluated as an optimization problem, since the objective is to maximize or minimize evaluation criteria, such as: adjusted coefficient of determination or R²aj (Equation 6), mean square error or MQE, which is defined as the quadratic mean of the difference between the estimated value and the actual value of the data (Equation 7). Besides that, there is the Cp statistic, which is a measure of the total mean square error for the regression (Equation 8).

$$\mathrm{R}^2 ajust = 1 - \left(\frac{\mathrm{n}-1}{\mathrm{n}-\mathrm{p}}\right)(1-\mathrm{R}^2) \tag{6}$$

$$\mathrm{MQE} = \frac{SQE}{(\mathrm{n}-\mathrm{p})} \tag{7}$$

$$\mathrm{C}p = \left(\frac{SQE(p)}{\sigma^2}\right) - n - 2\mathrm{p} \tag{8}$$

Since SQE is the quadratic sum of the error, σ^2 is the variance, R^2 is the determination coefficient, n is the number of variables and p is the number of samples.

4. METHODOLOGY

The flowchart of Figure 8 shows the steps performed in this work:

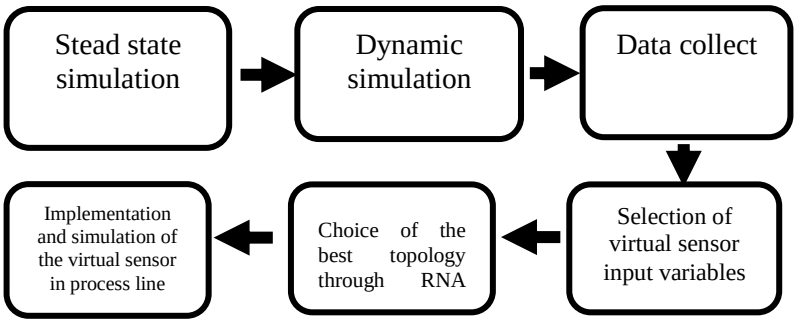


Figure 8. Methodology used in the work development for the construction of virtual sensors

5 MODELING AND SIMULATION OF ETHYLBENZENE PRODUCTIVE PROCESS

5.1 Production process of ethylbenzene

In the study route, ethylbenzene (EB) is produced in two CSTR reactors in series, from the reaction of ethylene (E) and benzene (B). In this process, the formation of the diethylbenzene compound (DEB), originated from the secondary reaction between the EB and the B occurs. The process consists of two reactors and two distillation columns. The output of the second process reactor is the feed stream from the first multicomponent distillation column, the compound being the "light switch" the benzene (> 99.9 mol%) recycled to the first process reactor. The second distillation column is fed to the base stream of the first column, providing an EB-rich (> 99.9 mol%) top stream and a DEB rich stream (> 99.9 mol%). This last recycle stream to the second reactor. Thus, the process proved to be interesting for the present work in view of the occurrence of transient effects in the high purity compositions of the top and bottom streams of the second distillation column.

The production of EB involves the liquid phase reaction between ethylene and benzene (Equation 9):

$$C_2H_4 + C_6H_6 \rightarrow C_8H_{10} \tag{9}$$

There are also some undesirable reactions, among which the main one is the formation of the DEB, according to Equation 10:

$$C_8H_{10} + C_2H_4 \rightarrow C_{10}H_{14} \tag{10}$$

In addition, DEB reacts with benzene and again forms EB, and it is possible to recycle the DEB that leaves the second reactor to increase production, Equation 11:

$$C_{10}H_{14} + C_6H_6 \rightarrow 2C_8H_{10} \tag{11}$$

Table 2 below shows the reaction kinetics, obtained in Luyben (2011). It is important

to point out that the activation energy of the undesirable reaction is greater than that of the desired reaction, so lower temperatures favor the formation of EB in the second reactor.

Table 2. Ethylbenzene reaction kinectics

	Reaction 1 (Equation 9)	Reaction 2 (Equation 10)	Reaction (Equation 11)
K	$1{,}528 \times 10^6$	$2{,}778 \times 10^7$	1000
E (cal/mol)	17 000	20 000	15 000
Concentration terms (kmol/m³)	$C_E C_B$	$C_E C_{EB}$	$C_B C_{DEB}$

Figure 9 shows the flow diagram of the ethylbenzene process with the size of the equipment and the values of the feed and outlet currents. The first reactor, with an operating volume of 200 m³, receives a feed stream of 630.6 kmol / h of pure benzene and another pure ethylene stream of the same molar flow. It operates at 434 K and 20 atm, and is cooled by 414 K steam in a coil, plus a cooling jacket with a 151 m thermal exchange area, totaling 602 m 2 of thermal exchange area required to remove 2.46 X 10^6 cal / s.

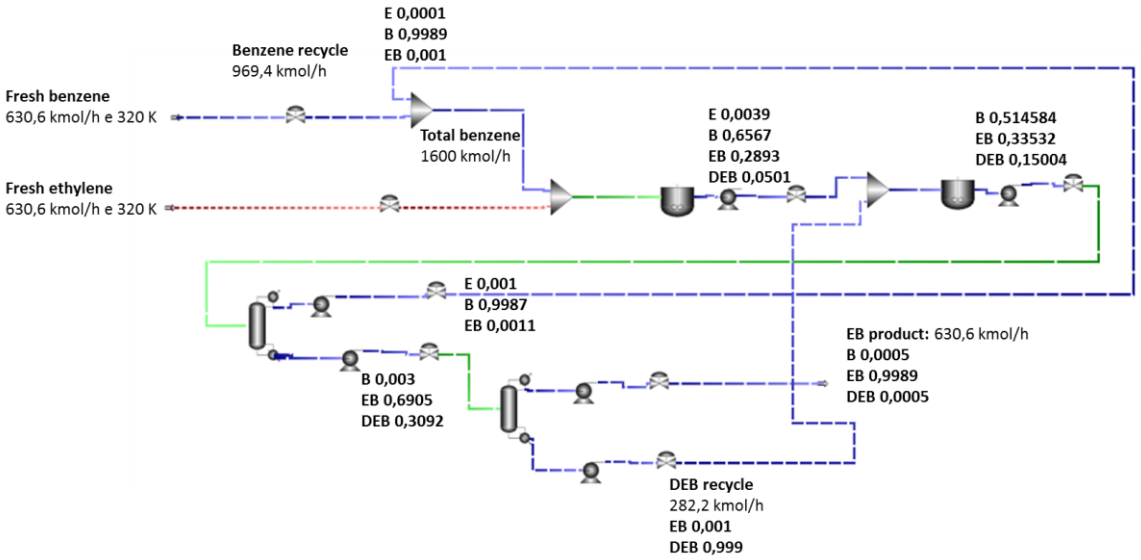


Figure 9. Ethylbenzene process flowsheet

The effluent from the first reactor is feed from the second, where practically all the DEB generated in the first reactor is converted to EB from the reaction with the benzene. In addition, the DEB rich stream (282.2 kmol/h) leaving the bottom of the second distillation column feeds the second reactor through a recycle stream, operating at 432 K and 19 atm. The effluent of this leaves at high pressures and temperatures.

The output of the second reactor is feed from first distillation column, which has 21 stages, with a reflux ratio of 0.774, with a thermal load of the 8.13 MW reactor and operating at a moderate vacuum of 0.3 atm. It has a base temperature of 390 K, which allows the use of a low-pressure steam (433 K, 4 atm) in the boiler. It is worth noting that the bubble points for B, EB and DEB are 353, 409 and 457 K, respectively; the significant differences justify the

use of a few dishes in the two columns and the low reflux rate. The second distillation column has 25 stages and a reflux rate of 0.661. It operates at a pressure of 0.1 atm, with a base temperature of 407 K, and produces a high purity distillate in EB (99.9 %mol) and has its base stream recycled to the second reactor.

5.2 Stead state simulation

The virtual sensor input data is preferably derived from simulation results, since data from physical sensors installed in the plant may not be reliable due to measurement problems such as mismatching and interference. (FORTUNA et al., 2007). In addition, steady-state simulation gives the engineer the knowledge of the initial conditions required for dynamic operation and requires knowledge of the behavior of the steady-state process for a good understanding of the dynamics of the compositions in a distillation column (MORAIS JR, 2015).

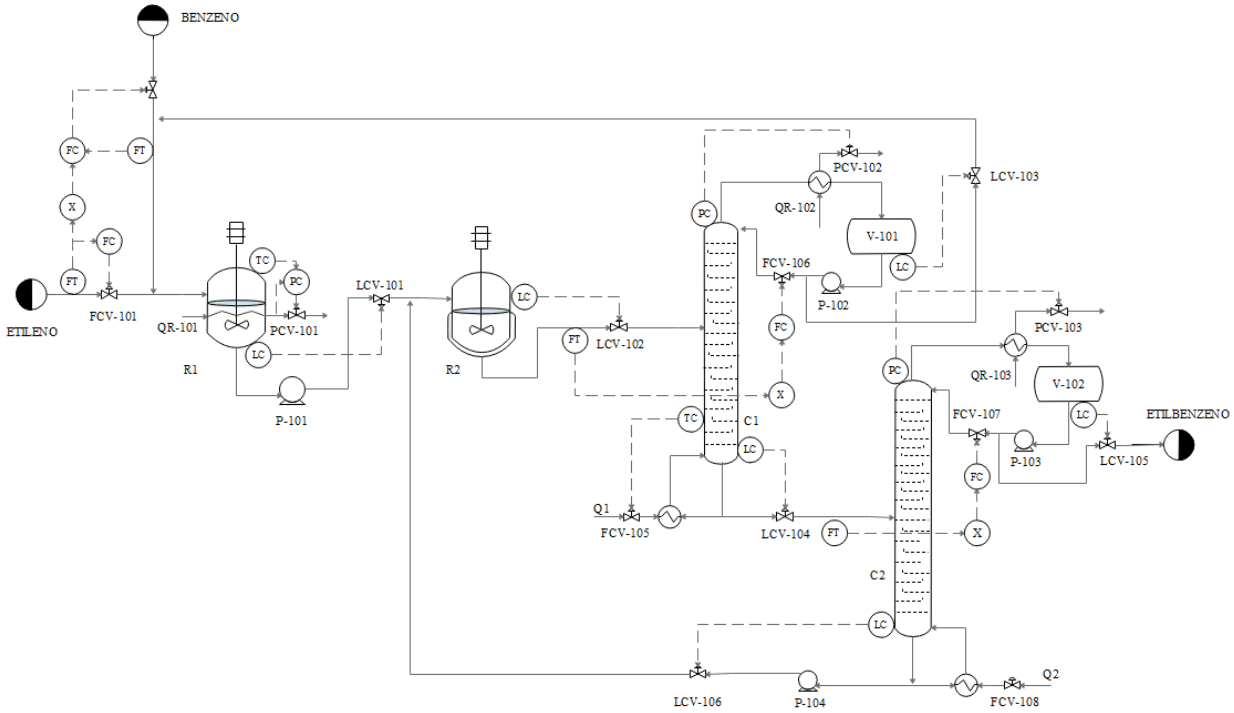
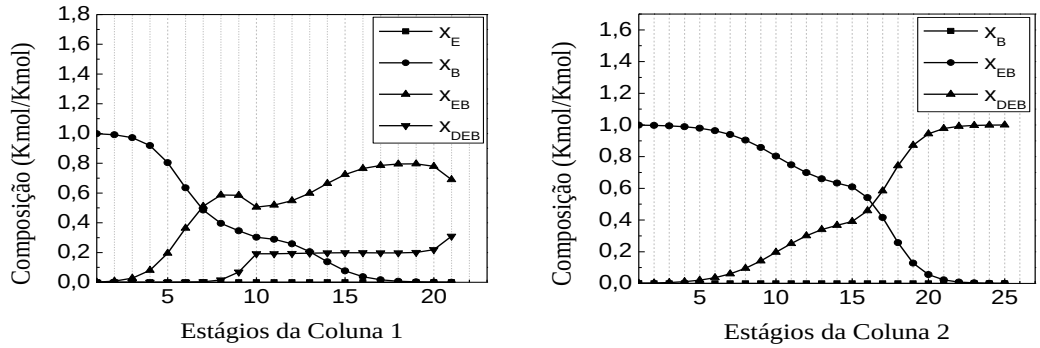


Figure 10. Ethylbenzene process dynamic simulation.

The productive process of the EB, shown in the flowchart of Figure 10, was simulated steadily in the Aspen PlusTM software, with the design data and initial conditions obtained in Luyben (2011). The two degrees of freedom required for the distillation process were fixed, with the following variables: base flow and reflux ratio. These variables were manipulated to fit plant data. The thermodynamic model used for calculations of physical properties was by Chao-Seader. All the results were compatible with those presented in the literature.

The graphs of Figures 11a and 11b show the composition profiles over the stages of columns C1 and C2. It will be seen, according to Figure 11a, that at C1 the benzene, xB, leaves in high purity at stage 1 (top). According to Figure 11b, ethylbenzene and diethylbenzene, xEB and xDEB, are in high purity in stages 1 and 25 (top and bottom) of C2, respectively.



11.a Column C1. **11.b Column C2.**

Figure 11. Molar composition profiles on the distillation columns C1 and C2

5.3 Dynamics Results

After performing and evaluating the simulation in steady state, the equipment was dimensioned and the conversion to the transient was made, whose simulation took place in Aspen Dynamics^{TM}, where perturbations were made in the following variables of C2: thermal load (Q2), flow rate Inlet (F2), top and bottom flow (B2) and (D2), as shown in Figures 12 and 13.

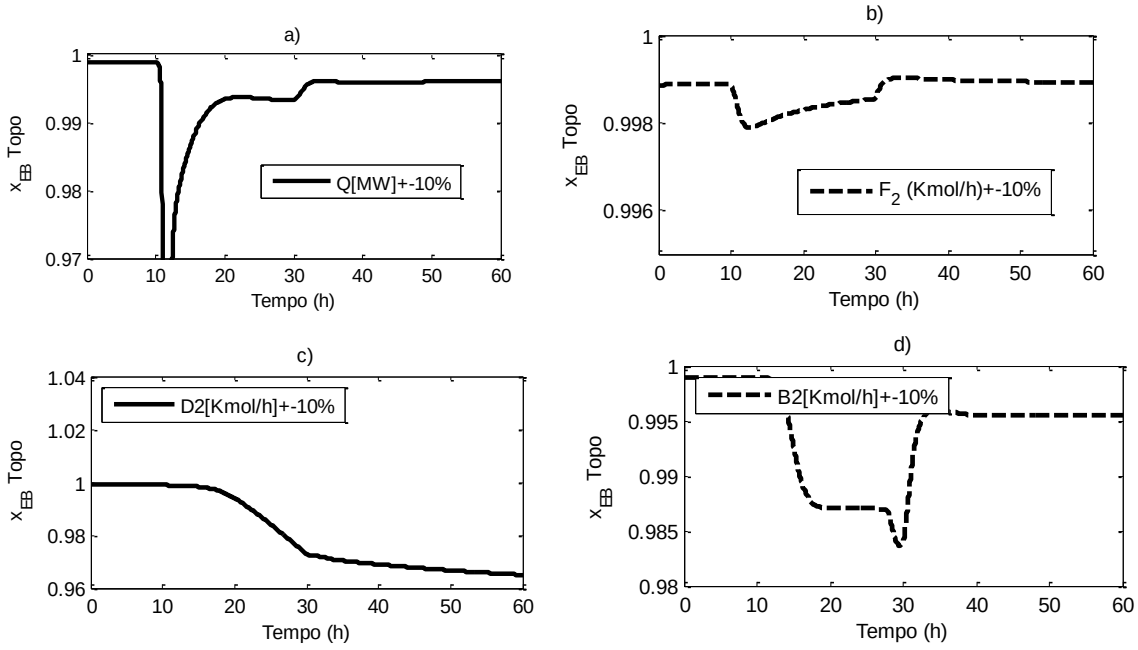


Figure 12. Composition responses of EB at the top for perturbations in C2.

According to graphs of Figure 12, the composition of ethylbenzene (xEB) is sensitive to changes in variables Q2, F2, D2 and B2, providing for this compositional response over considerable percentage increases. Likewise, it is found that for the same effected disorders, changes were made in the composition of diethylbenzene in base stream of column 2 (xDEB), Figure 13.

For the procedure of data collection necessary for selection of soft sensor input variables and neural network offline training, disturbances with ± 10% amplitude were inserted in F2, D2 and B2; And pseudorandom-random signals (PRBS) at Q2, with an amplitude of 5%, all at different times.

After assessing the responses to disturbances, it was noted that all previously selected variables exerted significant influence on the compositions. The set of 8 regressors, as shown in Table 3, was used to select variables from inference models of the EB and DEB compositions at the top and bottom of the second column, respectively.

Table 3. List of candidate variables for selection

OUTPUTS	INPUTS							
	u1	u2	u3	u4u	u5	u6	u7	u8
yEB (top) on column 2 (kmol/kmol)	Flow rate F2 (kmol/h)	Reboiler Duty Q2(MW)	Top Flow FD2 (kmol/h)	Bottoms Flow FB2 (kmol/h)	Reflux Flow R2 (kmol/h)	Stage 20 temperature T20 (K)	Stage 21 temperature T21 (K)	Stage 22 temperature T22 (K)

6 Variable Selection Results

After the use of the all possible regressions and stepwise regression methodologies to select secondary variables in inference models, the five best models presented in Table 4 were obtained. The two best models for the output variable (yEB- top) and their statistical evaluation criteria.

Table 4. Best sets of secondary variables

	All possible regressions			**Stepwise regression**	
Models yEB-top	**MQE**	**Cp**	**R^2 aju**	**MQE**	**R^2 aju**
[u1 u2 u3 u4 u5 u6 u7 u8]	0,000654	0,9970	0,8906	0,000654	0,8906
[u1 u2 u3 u4 u5 u7 u8]	0,000673	0,9967	0,8875	0,000673	0,8875
[u1 u2 u3 u4 u5 u6 u8]	0,000678	0,9967	0,8865	0,000678	0,8865
[u1 u2 u3 u4 u6 u7 u8]	0,000679	0,9967	0,8864	0,000679	0,8864
[u1 u2 u3 u4 u5 u8]	0,000684	0,9963	0,8851	0,000684	0,8851

Stepwise regression results confirmed those of all possible regressions, and similar values were obtained for the statistical evaluation criteria. No step of variable withdrawal was required for the stepwise method. It is worth noting that, when comparing the two methods, all possible regressions are recommended, given that it can find, in a global way and among all possibilities, the best regression equation according to the mentioned criteria. The stepwise methodology may distort the procedure by considering the dependencies between the regressors. The model containing the eight variables reached the best criteria: lower MQE and higher Cp and adjusted R^2. Finally, the following variables were selected: flow (feed, top,

base and reflux), besides the thermal load and the temperatures of stages 20, 21 and 22, all referring to column 2. Therefore, these variables exert influence on the leaving compositions of the interest compounds.

7 CONSTRUCTION AND IMPLEMENTATION OF VIRTUAL SENSOR

In order for the virtual sensor adequately supply the distillation column composition outputs, it is necessary to collect and divide input and output data and define input variables. In addition, neural network training is necessary, with steps to define the best network architecture and the best number of hidden layers to reach the defined objective (MORAIS JR, 2015).

7.1 RNA offline training

The values of the several variables were collected during 60h in the simulation stage. The variables selected in the previous step were defined as input variables of the neural network because they exert a greater influence on the output compositions to be inferred.

The activation function used was logsig, since it is the most used for the construction of ANNs, in addition to presenting a balance between linear and nonlinear performance (SANCHEZ, 2009). Since the logsig function has a normalization interval [0 1], the input signals were normalized for better inference performance since the inputs had different units. In addition, in the process of ANN learning, it is customary to divide the data between the training, validation and test steps. In this study, 70% of the data set were used for the training, 15% for the validation and 15% for the test.

Two types of architectures were tested: Elman's network and multilayer perceptron (MLP). It was observed that, in addition to MLP, it resulted in a lower error value between the actual and estimated output by the network, the training time for it was significantly lower - while an Elman network training lasted up to 30 min, MLP It took only 2 to 3 minutes to complete the training. Such behavior is justifiable, since the multilayer perceptron network can be trained to approach virtually any smooth and measurable function such as mass and energy balances. Unlike other statistical techniques, MLP does not make prior assumptions about data distribution. It can model highly non-linear functions and can be trained to generalize accurately when new usage data are presented (GARDNER and DORLING, 1998), and learn real-time models (online learning).

For this step, a computer was used an AMD PhenomTM II-X4-B95 3.0 GHz and with 8GB of RAM. The software used was Matlab®, through the Neural Network Toolbox, which provides functions and applications for the modeling of complex nonlinear systems that are not easily modeled with a closed form equation, in addition to supporting supervised learning.

Regarding the definition of number of neurons in the hidden layer, this is obtained experimentally, through a process of trial and error. In spite of this, a network with few neurons in the hidden layers is preferable because it presents better problem generalization power, that is, problems of overfitting or overfitting are reduced. However, the use of few neurons in the hidden layers can compromise the ability of the network that is sufficient to model and learn the data of complex problems (underfitting). Thus, it is paramount to find a

Proceedings of the XXXVIII Iberian Latin-American Congress on Computational Methods in Engineering P.O. Faria, R.H. Lopez, L.F.F. Miguel, W.J.S. Gomes, M. Noronha (Editores), ABMEC, Florianópolis, SC, Brazil, November 5-8, 2017..

number of neurons per layer that is ideal for generalizing input data without compromising learning (MORAIS JR, 2015 and HAYKIN, 2001).

7.2 ANN offline training

Thus, several amounts of neurons were tested in two or three hidden layers, as Table 5, since the use of a large number of hidden layers is not recommended. Each time the average error during training is used to update the synapse weights of the immediately preceding layer, it becomes less useful or imprecise.

Table 5. RNA topologies evaluated for virtual sensor construction

Number of Hidden Layers	Number of neurons
1	5, 10,..., 20
2	5, 10,..., 20

Finally, the training algorithm chosen was Levenberg-Marquardt (LVM), which, among other training algorithms, has a faster convergence. After all the tests to evaluate the choice of the architecture, the activation function, the training algorithm and the number of layers with their respective number of neurons, about 160 RNAs were trained. The best result, that is, the smallest error between the actual data and the estimated EB composition data at the top of the second distillation column was obtained with the topology [15 15 15]: one input layer and two hidden layers, with 15 neurons each, as shown in Table 6 below.

Table 6. ANN data with best offline result for composition inference

Topology	[15 15 15]
Training Algorithm	Levenberg-Maquardt
Architecture	Multilayer perceptron
Activation function	Logsig
Error performance	2,27 x 10-6
Number of iterations	78

Generally, regression analysis of the data is used to evaluate the relationship between two or more variables. It is a technique that allows visualizing the relation of a dependent variable to other independent variables (MONTGOMERY and RUNGE, 2012; HAYKIN, 2001). Thus, the adjustment of the data in the training, validation and test steps for this chosen

neural network is shown in Figure 14:

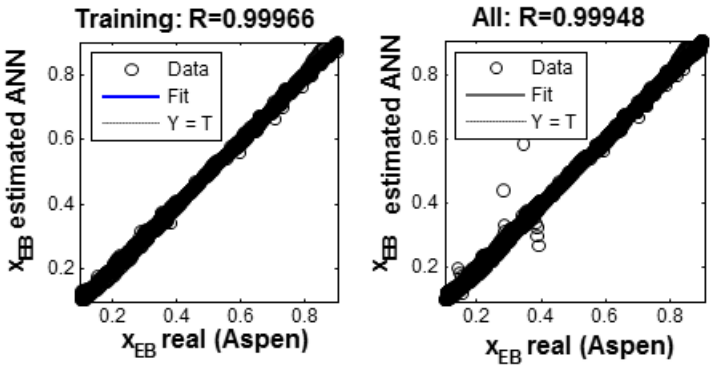


Figure 14 a, b. Data adjustmet at each stage of the neural network.

As can be seen in the graphs above, the estimates generated by the neural network model fit with the real data at a 45º slope, which provides a fitting conclusion to fit the experimental data. The graphs of Figures 15 (a) and (b), respectively, present the results of the XEB Virtual Neural Sensor training and validation data set, respectively, compared to data from the Aspen Dynamics^{TM} simulator.

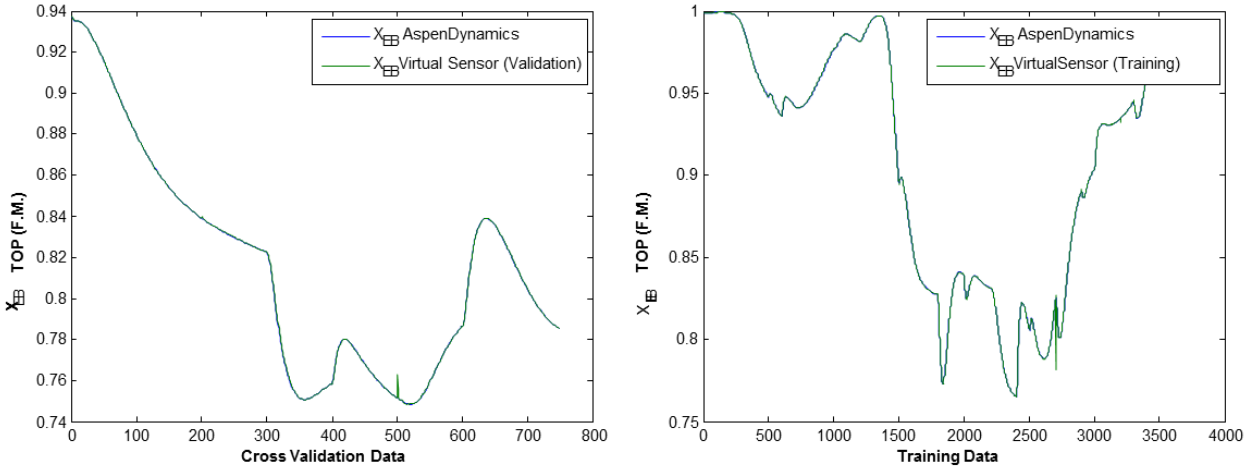


Figure 15. a) Neural Virtual Sensor of x_{EB} and simulator output (test); b) Neural Virtual Sensor of of x_{EB} and output of the simulator (cross-validation)

It is important to mention that this step was fed with a set of different data from which they were manufactured (15% of the total data), but without extrapolation of training limits. Thus, it is verified that the cross-validation data provide good results, which increase the generalization capacity of the developed neural virtual sensor. Already in Figure 15 the inference results were presented with all the data of the top composition of the EB in column 2 during 60h of simulation.

Therefore, the virtual sensors presented good results for inference of the high purity composition of the distillate from the second distillation tower. It is important to observe the

occurrence of several perturbations in the input variables, set of Data 1, which increases the robustness of the sensors and guarantees a good reliability of estimation.

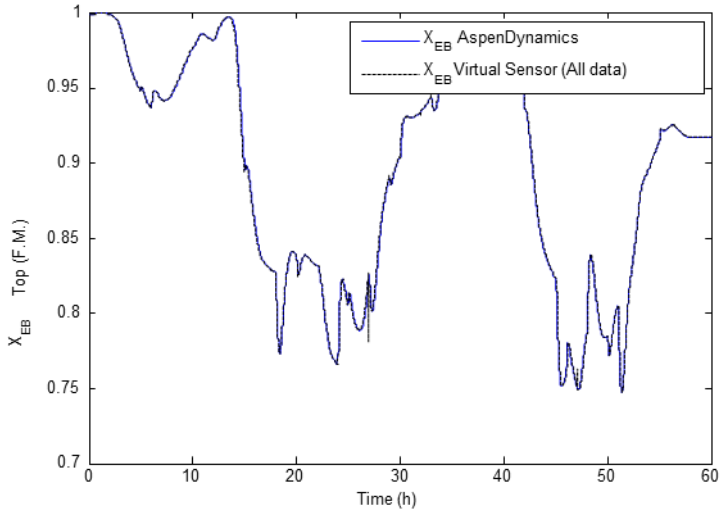


Figure 16. X_{EB} Dynamic Neural Virtual Sensor and simulator output

7.3 Inferential control implementation

This step consists of implementing, in the simulated process line by Aspen Dynamics, the best RNA chosen for inference as a measurement of EB composition at the top of the second distillation column (C2). For this purpose, a communication code block was generated between the virtual sensor in the Matlab® software and the plant. A scheme is shown in Figure 17.

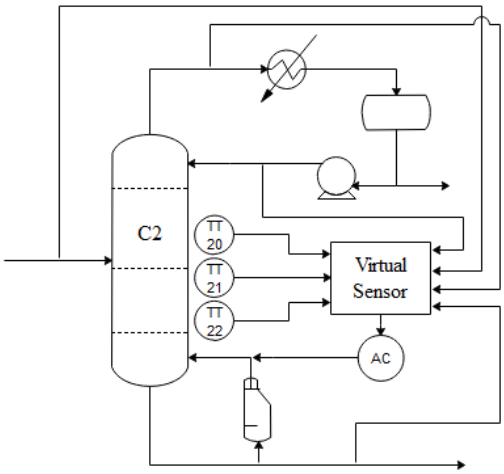


Figure 17. Composition control scheme through neural network measurement.

To evaluate the effectiveness of the implemented controls, perturbations were made in a large process disturbance, which is the fraction of benzene in the feed flow in -20% (in 10h) and + 20% (in 20h). Next, the behavior of the composition was evaluated: in open mesh,

when there is no control over x_{EB}; When the control is done indirectly, using the temperature of the most sensitive dish (T_{17}) of the column ($x_{EB\ TEMP}$); and using measurement information from the estimates provided by the neural virtual sensor ($x_{EB\ COMP}$). The results of control strategies is shown in Figure 18.a, and difference between the two control strategies is shown more clearly in details from Figure 18.b.

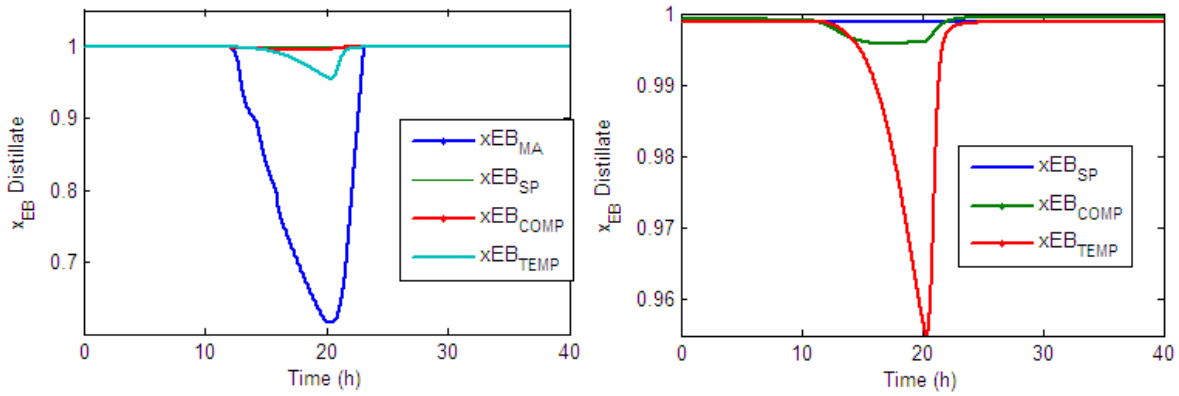


Figure 18 a, b. Composition control using different strategies.

As can be seen from Figure 18, a qualitative analysis of the composition in the open loop condition ($x_{EB\ MA}$) was not able to reestablish steady state condition the column. When comparing the temperature control of column 17, its effect evaluated in the composition of ethylbenzene ($x_{EB\ TEMP}$), and the composition via virtual sensor with ANN ($x_{EB\ COMP}$), it was observed that the control by $x_{EB\ COMP}$ was more effective, occurring with lower stabilization time and a lower percentage overshooting. In this way, the virtual sensor inferential control proved to be a viable alternative for implantation of the methodology in the industrial process of production of the ethylbenzene compound.

8. CONCLUSIONS

The ethylbenzene production process was studied, simulated under steady and transient conditions. The results obtained by the simulation were consistent with those obtained in literature (LUYBEN, 2011).

For the selection step of soft sensor input variables, some variables were previously defined that most influenced the ethylbenzene output composition at the top of the second distillation column. There were disturbances in the variables thermal load (Q2), inlet flow (F2), top and bottom flow (B2) and (D2). In addition to these, the reflux flow (R2) and the three temperatures of the column stages identified as the most sensitive were added to the initial set of regressors: T20, T21 and T22. Two methodologies were used for the selection of variables: all possible regressions and stepwise regression. Both were efficient, but it is recommended to use all the possible regressions, since it can find, in a global way and between all possibilities, the best regression equation according to the mentioned criteria. The input variables defined for the virtual sensor were all those previously mentioned.

In the construction of the virtual sensor, the use of the multilayer perceptron architecture (MLP) was defined, because it presented faster convergence and shorter training

time, compared to other architectures such as Elman Networks. After testing several combinations of layers and number of neurons, the neural network that obtained better performance was the one that had three layers, with 15 neurons in each, logsig activation function and Levenberg-Maquardt training algorithm, converging with an error of 10^{-6} in only 78 iterations. The adjustment between the experimental data and the data inferred by the network obtained an R^2 of 0.99948. Thus, the network was efficient and was able to comply with its virtual sensor function of the top composition of the EB in the second distillation column of the plant.

As for the application of the control, the control strategy that uses the measurement of the neural network was shown to be the most efficient occurring with lower stabilization time and lower percentage overshooting. In this way, the virtual sensor inferential control proved to be the viable alternative for implantation of the methodology in the industrial process of production of the ethylbenzene compound.

REFERENCES

Luyben, W. L., 2011. Design and Control of the Ethyl Benzene Process. *AIChE Journal*, v. 57, p. 655-670.

Luyben, W. L., 2013. Distillation design and control using Aspen simulation. 2nd ed., Copyright by John Wiley & Sons, Inc.

Haykin, S., 2001. Redes neurais: princípios e prática. Second edition. Bookman.

Fortuna, L.; Graziani, S.; Rizzo, A.; & Xibilia, M.G., 2007. *Soft sensors for monitoring and control of industrial processes*. London, UK: Springer.

Morais Jr, A. A., 2015. *Uso de sensores virtuais (soft sensors) para estimativa de impurezas em colunas de destilação de alta pureza*. PhD thesis, PPGEQ/Campina Grande.

Fortuna L., Graziani S., Xibilia M. G., 2005. Soft sensors for product quality monitoring in debutanizer distillation columns. *Control Engineering Practice*. Vol.13(4), 499-508.

Montgomery, D. C.; Runger, G. C., 2014. *Applied Statistics and Probability for Engineers*. 6th edition. New York: John Willey & Sons.

Gardner, M. W.; S. R. Dorling, 1998. Artificial neural networks (the multiplayer perceptron) – A review of applications in the atmospheric sciences. *Atmos. Environ.*, 32, 2627–2636.

Silva, I. N.; Spatti, D. H.; E Flauzino, R. A., 2010. *Redes neurais artificiais para engenharia e Ciências aplicadas: curso prático*. São Paulo, Editora Artliber.

Levenberg, K., 1944 A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Math*, Providence, v. 2, p. 164-168.