

Multiobjective Stochastic Optimization of Dividing-wall Distillation Columns Using a Surrogate Model Based on Neural Networks

C. Gutiérrez-Antonio^{a,*} and A. Briones-Ramírez^{b,c}

^aUniversidad Autónoma de Querétaro, Facultad de Química, Cerro de las Campanas s/n Col. Las Campanas, Querétaro, Querétaro, 76010, México

^bExxerpro Solutions, Av. del Sol 1B Interior 4B, Col. El Sol, Querétaro, Querétaro, 76113, México

^cInstituto Tecnológico de Aguascalientes, Departamento de Ingeniería Química, Av. López Mateos # 1801 Ote. Fracc. Bonagens, Aguascalientes, Aguascalientes, 20256, México

doi: 10.15255/CABEQ.2014.2132

Original scientific paper
Received: October 16, 2014
Accepted: December 2, 2015

Surrogate models have been used for modelling and optimization of conventional chemical processes; among them, neural networks have a great potential to capture complex problems such as those found in chemical processes. However, the development of intensified processes has brought about important challenges in modelling and optimization, due to more complex interrelation between design variables. Among intensified processes, dividing-wall columns represent an interesting alternative for fluid mixtures separation, allowing savings in space requirements, energy and investments costs, in comparison with conventional sequences. In this work, we propose the optimization of dividing-wall columns, with a multiobjective genetic algorithm, through the use of neural networks as surrogate models. The contribution of this work is focused on the evaluation of both objectives and constraints functions with neural networks. The results show a significant reduction in computational time and the number of evaluations of objectives and constraints functions required to reaching the Pareto front.

Key words:

neural networks, multiobjective genetic algorithms, surrogate model, dividing-wall distillation column

Introduction

Chemical processes have a multiobjective nature, since normally there are several objectives in conflict with each other; which are also restrained to requirements, physical or economical limitations. Since in multiobjective problems, the objectives are in conflict with each other, a simple solution is not desirable or sometimes even not feasible. For this kind of problem, a set of optimal solutions that represents the best trade-off between these objectives is the goal. These optimal designs can be achieved by means of Pareto front, which is a set of optimal non-dominated solutions¹. In this way, the Pareto front allows having not just one optimal solution, but a set of optimal solutions that represents the best compromise between the objectives in consideration.

Evolutionary algorithms have been recognized to be well suited for multiobjective optimization,

because of their capability to evolve a set of non-dominated solutions distributed along the Pareto front². One of the most popular multiobjective optimization algorithms is the Non-dominated Sorting Genetic Algorithm, NSGA-II³, which is a very robust tool and it is easy to implement. However, the principal disadvantage of genetic algorithms, and its variants, is the large amount of computational time that is often required for multiobjective optimization of industrial operations⁴; this fact without considering if the evaluation of the objective function is computationally expensive. This has led to the development of new strategies or combination of strategies to reduce the required computational time; basically, these strategies are classified as those that modify the parameters of the algorithm, and those using surrogate models.

In the first type, key operators of the evolutionary algorithm are modified in order to give less randomness to the selection and generation of the individuals; the idea behind is to incorporate information about the problem, that can help to improve the search process. In general, the use of modified op-

*Corresponding author: Dra. Claudia Gutiérrez-Antonio, e-mail: claugtez@gmail.com; claudia.gutierrez@uaq.mx, Phone: (+52) 442 192 1200; Mobile: (+52) 442 329 93 48

erators in the evolutionary algorithms^{5–7} allows having a reduction in the computational time, without losing complexity in the model of the problem to solve. However, the modification of the operators must be realized in every strategy, which requires profound knowledge and skills. Also, these approaches have limited contributions to decrease the computational time when the evaluations of objective function are computationally expensive; especially if integer and continuous variables are involved.

On the other hand, the use of surrogate models allows using the same evolutionary approach without further modification, but the changes are made in the evaluation of the objective function. Thus, the objective function can be replaced for a previously known reduced model, or there is another optimization process nested in order to find the best model that captures all the complexity of the original problem. The contribution of this work relies in this kind of approach; therefore, next we will present a review of some related works.

Poloni *et al.*⁸ presented a strategy that combines several optimization techniques in order to solve a multiobjective optimization problem. The techniques used were genetic algorithms, neural networks and gradient based search. Nevertheless, the use of the optimization strategies is sequential and lacking in integration; also, deep knowledge of the three strategies is needed.

Nain and Deb⁹ proposed a different approach using genetic algorithms and neural networks. Their strategy consists of training a neural network using the results of the genetic algorithm. After generating the approximate functions, they fixed an a priori number of generations; in these generations, the approximate functions are used instead of the original functions. The principal limitation in this approach is the adequate estimation of this fixed number of generations, where the approximated functions must be used.

Later, Gaspar-Cunha and Vieira¹⁰ adapted the approach of Nain and Deb⁹ through two variations. In the first one, they just change the multiobjective optimization strategy; Nain and Deb used the NSGA-II³, while Gaspar-Cunha and Vieira used the Reduced Pareto Set Genetic Algorithm with Elitism, RPSGAe¹¹. In the second variation, a percent of the total individuals was simultaneously evaluated with the exact and approximated functions. The main contribution of their work was the introduction of an error variable, which recorded the error in the approximation of the neural network. Nevertheless, real data or the use of the exact function is required in order to have all the information for error calculation. Also, the handling of constraints was not considered.

In 2007, Zhou *et al.*¹² developed a strategy that combines global and local surrogate models to accelerate evolutionary optimization. Their framework uses hierarchical surrogate models constructed by online learning, and in this way replacing the computationally expensive objectives functions. This strategy employs data-parallel Gaussian process, memetic search, and Lamarckian evolution. However, deep knowledge of the three strategies is needed.

Recently, Sreekanth and Datta¹³ developed two types of surrogate models based on genetic programming and modular neural network; these models were coupled to the Non-dominated Sorting Genetic Algorithm, NSGA-II³. The application of these strategies for their particular problem reveals that the use of genetic programming generates less uncertainty than the modular neural networks.

In general, the use of surrogate models allows having a reduction in the computational time; however, the most important step is the estimation of the approximated functions, since it has to capture all the important information of the exact function. In addition, the frequency of use of the approximate functions is another key parameter to obtaining the best performance of the strategy. It is worth mentioning that, to our knowledge, the use of approximated functions for estimation of the constraints has not been reported.

Now, in Chemical Engineering, surrogate models have been used for modelling and optimization of conventional chemical processes, due to the high complexity and non-linearity of the involved models. In particular, the great ability of neural networks to capture complex models is well known; due to this, neural networks have been used to model and optimize conventional chemical processes in different applications such as chaotic chemical reaction systems¹⁴, crude distillation units¹⁵, large-scale reaction systems¹⁶, process synthesis¹⁷, conventional distillation sequences¹⁸, syngas generation and treatment¹⁹, integrated gasification combined cycle²⁰, biodiesel production²¹, and power plant design²². However, the development of intensified processes has brought about important challenges in modelling and optimization, due to the more complex structure and relation between all design variables, with respect to conventional chemical processes. We can define process intensification as a strategy for making dramatic reductions in the size of a chemical plant for a given production objective; these reductions can come from shrinking the size of individual pieces of equipment, and also from cutting the number of unit operations involved²³. Among the intensified processes, dividing-wall columns represent an interesting intensified process for the separation of fluid mixtures, allowing savings in

energy costs, investments costs and space requirements in comparison with conventional distillation sequences.

In this work, we propose the optimization of dividing-wall distillation columns using a multiobjective genetic algorithm with constraints handling²⁴, through the use of neural networks as surrogate models. The evaluation of the objective and constraints functions is performed with neural networks. Our proposal considers the use of surrogate functions, neural networks, which can be used in any optimization strategy; until now, the combination of these tools was focused on the use of genetic algorithms as surrogate models for neural networks optimization strategies^{25–28}. Therefore, our contribution involves the use of a simple structure for the neural network in order to be used as surrogate model for the optimization of really complex and non-linear systems: dividing-wall distillation column. Besides, to the author's knowledge, this combination of strategies has not been proposed before.

We tested the proposed approach in the optimization of dividing-wall columns, considering the complete model of the column, set of MESH equations, which is highly non-linear involving integer and continuous variables. The results show a significant reduction in the number of evaluations of the original objective and constraints functions, as well as in the computation time required to reaching the Pareto front.

Artificial neural networks

Artificial neural networks are a biologically inspired computational model formed from hundreds of single units, artificial neurons, connected through coefficients, weights, which constitute the neural structure²⁶. The neural networks are a very powerful tool, since with relatively simple functions they can estimate the behavior of very complex problems. There are several types of neural networks, such as Radial basis function network, Hopfield network, neuro-fuzzy networks; the feedforward neural network is the first and most simple model, since the information just flows in one direction. Among the feedforward networks, the perceptron is the simplest kind. However, the power of this simple network should not be underestimated, since there is formal proof that the simplest neural network, the multilayer perceptron with two hidden layers, is theoretically sufficient to model any problem²⁹.

The crucial points in the construction of an artificial neural network are the selection of inputs and outputs, the architecture of the neural network (the number of layers and nodes in each layer), and finally, the training algorithm¹⁰. In this approach,

the inputs and outputs are already defined; since they are the objectives and constraints functions defined in the optimization problem. Then, the architecture and training algorithm must be defined.

Multiobjective genetic algorithm with constraints

In terms of multiobjective optimization, when a minimization takes place and the algorithm reaches a point where there is no feasible vector that can decrease the value of one objective without simultaneously increasing the value of another objective, that point in the search space is said to be the Pareto optimum. By definition, we can say that one point $\bar{z}' \in \mathfrak{S}$ is a Pareto optimum if for each $\bar{z} \in \mathfrak{S}$:

$$\bigwedge_{i \in I} (f_i(\bar{z}) \geq f_i(\bar{z}^*)) \quad (1)$$

Or at least there is some $i \in I$, where I represents the set of objective functions to optimize, that:

$$f_i(\bar{z}) > f_i(\bar{z}^*) \quad (2)$$

Then, we define that \bar{z} dominates \bar{w} when $f(\bar{z}) < f(\bar{w})$, if $W \subseteq \mathfrak{S}$ and $\bar{w} \in W$ if no $\bar{z} \in W$ dominates \bar{w} , we say that \bar{w} is not dominated with respect to W . As established by Mezura-Montes¹, the set of solutions which are not dominated and optimums of Pareto integrates the Pareto front. The Pareto front is integrated by all optimal points, which represent the best trade-off between the objectives of interest, including its extreme values and all points between them.

The construction of the Pareto front is made through a multiobjective genetic algorithm with constraints²⁴. This strategy is based on the NSGA-II³. Moreover, the constraints are handled with another multiobjective optimization technique, which guides the search of the genetic algorithm using the concept of non-dominance³⁰. The used approach is described next. The complete population is divided into subpopulations according to the number of constraints satisfied. Then, in a generation the best individuals are those that meet the n constraints; these individuals are followed by those that satisfy $n-1$ restrictions, and so on. In each subpopulation, the individuals are ranked using the value of the fitness function calculated with the algorithm based on NSGA-II; also, the minimization of the difference between the required and obtained constraints vectors is considered as another objective function. Thereby, the dominance inside each subpopulation is calculated using the following equation:

$$\text{dominance } \{ \text{objective}_1, \text{objective}_2 \dots \text{objective}_n, \\ (\text{constraint } s_{\text{required}} - \text{constraint } s_{\text{obtained}}) \} \quad (3)$$

From equation 3, it is clear that the calculation of dominance includes all the objectives of interest, along with another objective associated to the fulfillment of the constraints of the problem. The idea behind this is to minimize not just the objectives, but also the value of the restrictions unfulfilled. The most important characteristic of this approach to handling the constraints is that no additional parameters or tuning of the algorithm is required.

Proposed approach

The approach presented here allows decreasing the required computation time of stochastic strategies, where the evaluation of objective and/or constraint functions is computationally expensive, since neural networks are used as surrogate model. This approach is developed for a multiobjective genetic algorithm with constraints²⁴; however, the proposed approach can be implemented in any multiobjective or simple algorithm (stochastic or deterministic), where the evaluations of the objectives and constraints functions are computationally expensive. The optimization strategy used in this work is coupled with Aspen Plus processes simulator for the evaluation of both objectives and constraints. Aspen Plus simulator includes many models of different unit operations; however, there is no model available for dividing-wall columns. Due to this, we simulate the dividing-wall column as a Petlyuk sequence, since these configurations are thermodynamically equivalent when there is no heat transfer across the wall. The simulation of dividing-wall columns is computationally expensive due to the elevated number of interconnection streams. In our previous contribution²⁴, it was found that 95 % of the total time required for the optimization with a multiobjective genetic algorithm with constraints handling was consumed in the evaluation of objectives and constraints in a process simulator. Therefore, this case is appropriate for the use of surrogate models.

In order to reduce the computation time required to obtain the Pareto front, we propose the use of artificial neural networks, which creates approximate objective and constraint functions, based on training of individuals evaluated with the process simulator. The approximate functions are computationally less expensive, in the order of milliseconds in comparison with the time required for the evaluation of the original objective and constraints functions, in the order of seconds or even minutes. It is worth mentioning that the evaluation of objective and constraints functions is computationally expensive since they are performed in a commercial

process simulator, considering a complete equilibrium model.

The integration of the neural networks with the optimization strategy developed previously is presented in Figure 1; in this flowsheet, the rectangles in bold lines represent the original strategy, while the rectangles with dashed lines show the addition of the neural networks. As can be seen, the integration of this new approach is very simple and clear.

Throughout the optimization procedure, the multiobjective genetic algorithm generates, reproduces, and mutates all individuals, both existing and new; also, the dominances of all of them are calculated. The difference lies in the evaluation of the objective and constraints functions. From here, we called complete functions to those involved in the original functions (simulation in Aspen Plus process simulator), and estimated functions to those generated by the neural network; it is worth mentioning that objective and constraints functions are considered in both complete and estimated functions.

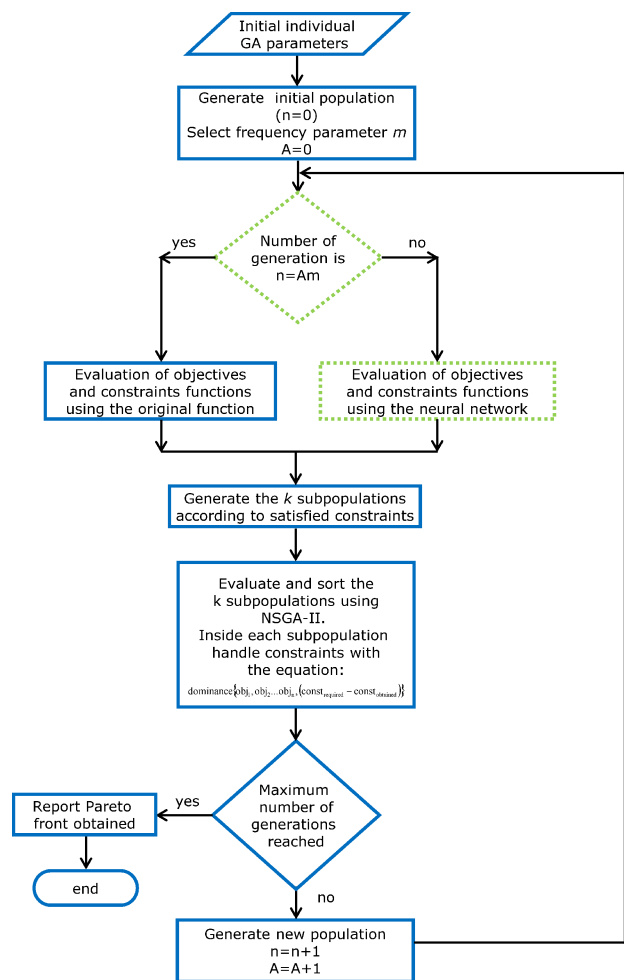


Fig. 1 – Flow diagram of the coupling of the speeding up approach with the optimization strategy

In the initial generation, $n = 0$, the evaluation of objectives and constraints is made with the complete functions (process simulator). The obtained results are used to create the individuals of the first generation, but also to train the neural network; in this way, we have estimated functions to evaluate objectives and constraints. Since the results of generation 0 allows having estimated functions, then a criterion to decide when to use the complete or the estimated functions must be defined. For this, we propose the frequency parameter, m , which determines how often the complete functions are used to evaluate the objectives and constraints of the optimization problem. The values of this frequency parameter can be 2, 3, 4 ... m_{\max} , since the value of 1 implies that the estimated functions are never used. In addition, it is important to point out that the maximum value of this parameter must be less than the number of generations; otherwise, the original functions are never used. The maximum value for the frequency parameter is then related with the retraining of the neural network, the complexity of the problem, and, indirectly, with the number of generations. One can expect that, as the complexity of the problem increases, the retraining of the neural network should increase also. Moreover, in the genetic algorithms, the number of generations increases as the problem is more difficult. Therefore, we decided to propose a simple relation between the complexity of the problem and the maximum value frequency parameter, related through the number of generations:

$$m_{\max} = \text{round}[\sqrt{n}] \quad (4)$$

Where n is the number of generations. In this way, as the complexity of the problem increases, the frequency of network training also increases. The value of this parameter goes from 2 to m_{\max} , where this last value depends the number of generations. We can then choose a moderate acceleration, $m = 2$, an aggressive acceleration, $m = m_{\max}$, or something between them, $m = (m_{\max} + 2)/2$. It is important to note that even if the minor value of m is used, 2, the number of evaluations of complete function is reduced by 50 %; this implies a considerable reduction in the computation time required for the optimization.

In order to use the frequency parameter properly, we introduce a counter A , which is an integer parameter with values from 0, 1, 2 ... q . Thereby, from generations 1 to $Am-1$ all individuals are evaluated with the estimated functions, as constraints and objectives; the Am generations are evaluated with the complete functions, which results are also used to train the neural network. In this way, during the Am generations, the estimation of the neural

network is improved with the data generated by the simulator (complete model); therefore, at the end of the optimization process, the error presented in the estimation of the neural network for both objectives and constraints is practically negligible, even when we are using a simple model of the neural network.

In summary, the use of estimated and complete functions for evaluation of objectives and constraints is switched every m generations; where m is the frequency parameter. In this way, we reduce the computation time, but we are still solving the original problem. The parameter m determines the frequency of use of the complete objectives and constraints functions, both for the optimization procedure and for the training of the neural network.

There are several types of artificial neural networks, and, of course, previous analysis and performance validation must be made before selecting the neural network for each problem. However, as mentioned previously, there is formal proof that the simplest neural network is theoretically sufficient to model any problem²⁹. Considering this, we used the simplest neural network in this work; nevertheless, any other neural network can be used in another problem.

Problem statement

The proposed approach was tested using as case of study the multiobjective optimization of dividing-wall columns. The Pareto fronts constructed with the multiobjective genetic algorithm with constraints, GAMOC, and with neural networks, GAMOC-ANN, were compared. In this section, first we describe the dividing-wall columns and its optimization problem. After that, the details about the parameters of the genetic algorithm and the neural network are presented, along with information of the mixtures to be separated in the dividing-wall columns.

Dividing-wall columns

Distillation is the most used separation process in chemical industries; however, a principal disadvantage of this technology is its high energy consumption. In order to overcome this drawback, thermally coupled distillation sequences have been proposed; these intensified schemes reduce significantly the energy consumption. The dividing-wall column, DWC, is the most applied thermally coupled distillation column, since it reduces operating costs by 35 %, investment costs by 25 %, and also the requirements of space by 40 %, in comparison with a conventional column sequence³¹; this makes DWC into something that corresponds with the present day idea of sustainable process technology³². In addition, it is important to mention that

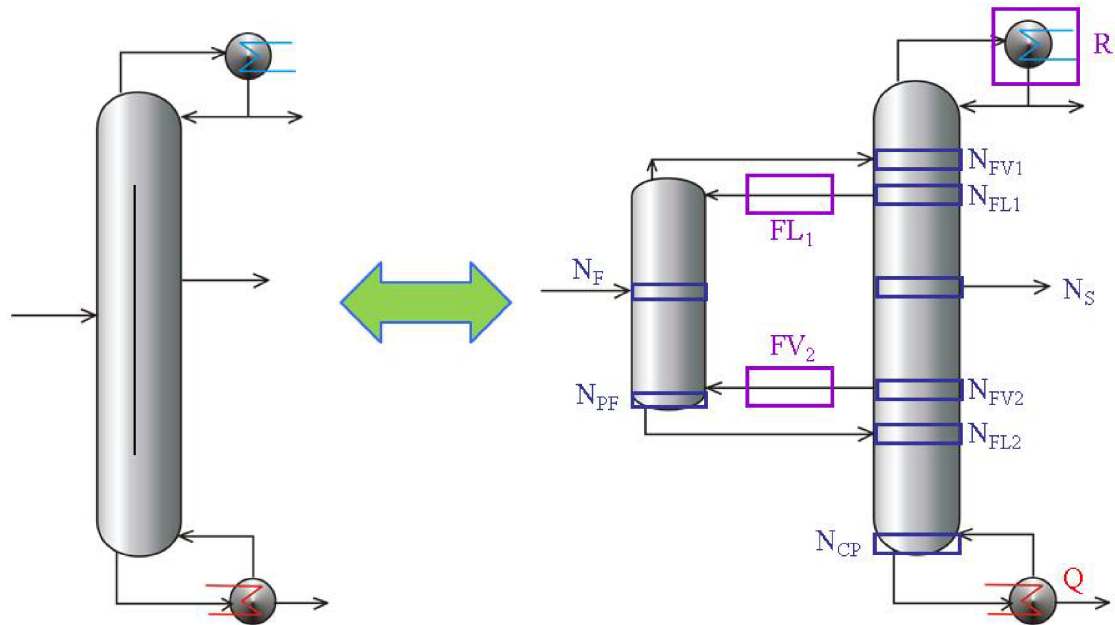


Fig. 2 – Design variables of the dividing-wall column for a ternary separation

these savings can be obtained without losing purity in the product streams. Dividing-wall columns have been widely studied for the separation of petrochemical mixtures³³, light hydrocarbons³⁴, azeotropic mixtures^{35,36}, bioethanol^{37,38}, biobutanol³⁹, and biojet fuel⁴⁰; moreover, this scheme has shown to be feasible for the performance of the reactive distillation process for the production of biodiesel⁴¹, mixed acid esterification and isopropanol dehydration⁴², ethyl acetate⁴³. This last work was performed in a pilot plant. Also, research efforts have been dedicated to the development of synthesis methodologies for the use of dividing-wall columns in multicomponent mixtures⁴⁴, multiple dividing-wall columns optimal design⁴⁵, and retrofitting of conventional distillation columns to dividing-wall columns⁴⁶. In 2010, more than 90 applications in production scale were known⁴⁷.

All the aforementioned clearly indicates the great importance of dividing-wall distillation columns as an efficient separation technology. Therefore, it is very important to have reliable optimization strategies that allow obtaining optimal designs with complete models, but with reduced computation time. The results can lead to the development of optimal design methodologies, which can be used as the base for the development of other strategies for azeotropic and reactive cases. In order to reach this objective, we propose the use of neural networks as surrogate models to estimate both objectives and constraints in the optimization of dividing-wall columns. According to the literature, the optimization problem of dividing-wall columns has been addressed by several researchers using different strategies as genetic algorithms^{33,36,40,45,48}, differ-

ential evolution^{52,53}, simulated annealing^{54,55}, response surface methodology⁵⁶, mathematical programming^{57–61} using complete models or shortcut methodologies. However, the alternate use of surrogate models with the complete model in Aspen Plus has not been proposed.

Optimal design of dividing-wall column

The dividing-wall column, DWC, is thermodynamically equivalent to the Petlyuk sequence, assuming that the heat transfer across the dividing wall is negligible; therefore, it can be represented as shown in Figure 2. From this figure, we observe that, for a ternary mixture, the optimal design of a dividing-wall column implies the calculation of 5 integer variables and 5 continuous variables, shown in Table 1. In addition, the purities and recoveries in each product stream must be reached (distillate, side and bottom streams); these are the constraints of the system. Moreover, there are three variables in competition with each other: number of stages on both sides of the wall, named here as pre-fractionator, N_{PRE} , and main column, N_{MAIN} , and the heat duty, Q , Figure 2. These variables are in competition because the minimization of one generates the opposite effect on the other, and vice versa. Thus, for a ternary mixture we have 3 objectives, 6 constraints, and 10 manipulated variables, integer and continuous. This optimal design problem can be expressed as:

$$\begin{aligned} \text{Min}(Q, N_i) &= f(R, F_k, N_{in,k}, N_{out,k}, N_F, N_S, N_i) \\ \text{subject to} & \\ & \bar{y}_k \geq \bar{x}_k \end{aligned} \quad (5)$$

Table 1 – Objectives, constraints and design variables of dividing-wall columns

	Dividing-wall column
Objectives	Number of stages in prefractionator Number of stages in main column Heat duty of the sequence
Constraints	Purity of component A in distillate stream Purity of component B in side stream Purity of component C in bottom stream Recovery of component A in distillate stream Recovery of component B in side stream Recovery of component C in bottom stream
Integer design variables	Number of stages in main column Location of feed stage Location of side stream stage Location of superior interconnection stream Location of inferior interconnection stream
Continuous design variables	Reflux ratio Interconnection liquid flow stream FL1 Interconnection vapor flow stream FV2 Distillate stream flow Side stream flow

Where R is the reflux ratio, F_k is the interconnection stream flow k , $N_{in,k}$ and $N_{out,k}$ are the number of stages where the interconnection flow k enters and exits, N_F is the feed stage of the sequence, N_S is the side stream stage, N_i is the number of stages in the column i . In addition, the constraints are the required and obtained purities and recoveries, vectors x_k and y_k respectively. In this work, the classical scheme of the dividing-wall column is considered, thereby the equality in the number of stages in the prefractionator and the section between the interconnection stages in the main column is considered as another constraint.

Parameters of the strategies and selected mixtures for the dividing-wall columns

In order to obtain the Pareto fronts for the single dividing-wall column, we established the following set of parameters.

For the multiobjective genetic algorithm, we used 50 generations for the dividing-wall columns of 1000 individuals each. It is important to mention that the number of generations and individuals is kept constant for the stochastic strategy with and without surrogate models; in this way, we can compare the performance of both approaches. The number of generations and individuals results from a tuning process. In this process, we varied the number of generations and the number of individuals in different combinations until we found the number of these parameters from which there was no significant change in the value of the objectives; in other words, that the variation in the value of the objectives function was just due to the randomness of the

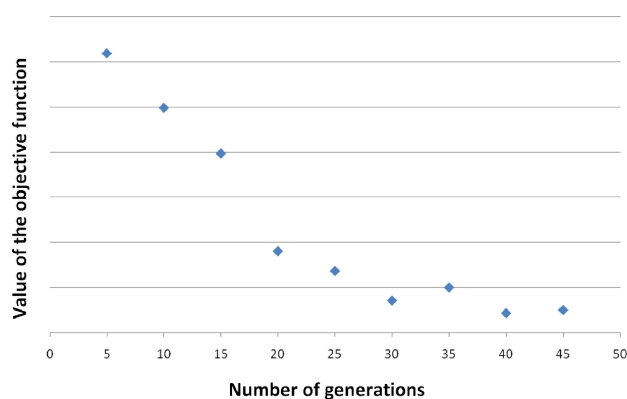


Fig. 3 – Convergence plot for an optimization problem with a fixed number of individuals

algorithm. For instance, Figure 3 shows the typical graphic used to determine the number of generations in an optimization problem, for a fixed number of individuals. In this graph, it is clear that beyond generation 45 there was no significant variation in the value of the objective function; so, for that case, 40 generations is the best value, given a fixed number of individuals. Nevertheless, it is important to mention that these parameters cannot be applied for the optimization of different chemical processes. In fact, parameters must be found for each chemical process that you want to optimize. In our experience, the number of individuals and generations tends to increase as the complexity of the problem increases (because the objectives increase or the number of constraints rises).

For the neural network, we chose a perceptron with two layers and eight cells in each layer. The functions included were sigmoid and linear, while the training was supervised with 30 cycles each one. We chose this simple neural network based on the conclusions provided by Kurková²⁹; but also, we tried to avoid generating another optimization problem nested for the selection of the best neural network. Considering the number of generations chosen for the genetic algorithm, we decided to make a moderate acceleration of the strategy; the frequency parameter was then chosen as 5 for the dividing-wall distillation columns.

The dividing-wall columns separate the following mixtures: n-pentane – n-hexane – n-heptane (M1), and methanol – n-butanol – methyl formate (M2). For both mixtures, thermal condition of feed is saturated liquid, while the recoveries and purities of key components in each product stream are [98 %-98 %-98 %] and [98.7 %-99 %-98.6 %] for mixtures M1 and M2, respectively. In order to compare optimization performance of GAMOC-ANN and GAMOC we employed the same initial designs, thermodynamic model, and convergence options for the simulation for both mixtures and procedures.

Discussion of results

In this section, we present the Pareto fronts of the dividing-wall distillation columns for the selected mixtures. The Pareto fronts were obtained using the multiobjective genetic algorithm with constraints, GAMOC, which considered just the use of the complete model in the process simulator. In addition, Pareto fronts were also generated with the same strategy with neural networks as surrogate models, GAMOC-ANN. A comparative analysis of the evolution of both Pareto fronts without, GAMOC, and with surrogate models, GAMOC-ANN, is presented next. It is worth mentioning that further analysis of the optimal designs themselves that integrated the Pareto front was not the objective of this work.

The Pareto fronts of the dividing-wall column were obtained by manipulating 10 variables, subject to the fulfillment of 6 constraints in the minimization of 3 objectives, Table 1. Pareto fronts show the three variables in competition as defined previously: number of stages in prefractionator, number of stages in main column, total heat duty. The designs that integrate the Pareto front represent the best trade-off between these variables, which are in competition. Since three objectives are considered, we present the results in 3D graphics.

The evolution of Pareto fronts of the dividing-wall column generated using GAMOC and GAMOC-ANN for mixture M1 is presented in Figure 4. In this fig-

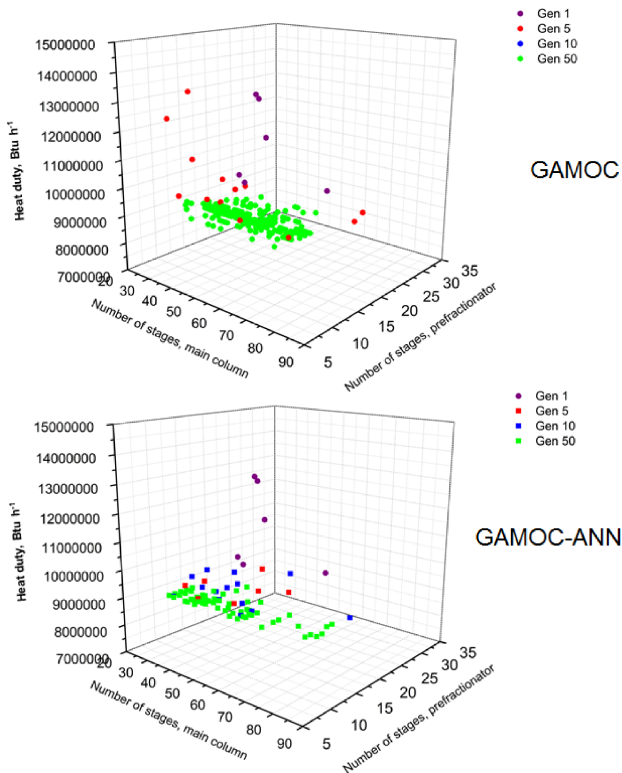


Fig. 4 – Evolution of the Pareto front of single dividing-wall column for mixture M1

ure, each series of dots in different color represent a different generation, beginning at generation 1. From these figures, we observe how the Pareto fronts evolve as the optimization proceeds; however, the followed trajectory is completely different when surrogate models are used. It is clear that with GAMOC-ANN, in generation 5, the individuals are more compacted in a region of lower energy, with respect to the individuals obtained in GAMOC in the same number of generations. From these figures, we also observe that the use of the neural networks allows reaching the Pareto front faster than the optimization strategy alone. Moreover, at the end of the optimization, in generation 50, the Pareto front of GAMOC is very close to the Pareto front generated with the surrogate model, GAMOC-ANN, but it is not there yet. Using surrogate models, 11,000 evaluations of the complete functions were required, in comparison with the 50,000 needed by GAMOC. Since the neural network is trained every time the complete model is used, the neural network improves the estimation of the objectives and constraints as they approach the optimal zone. Another important characteristic is that, in spite of the use of estimated functions, the Pareto front had maintained its diversity. Therefore, the use of surrogate models reduces significantly the number of evaluations of the complete objective functions, and the Pareto front is reached much faster. A similar behavior is observed for mixture M2 with and without the use of surrogate models, Figure 5.

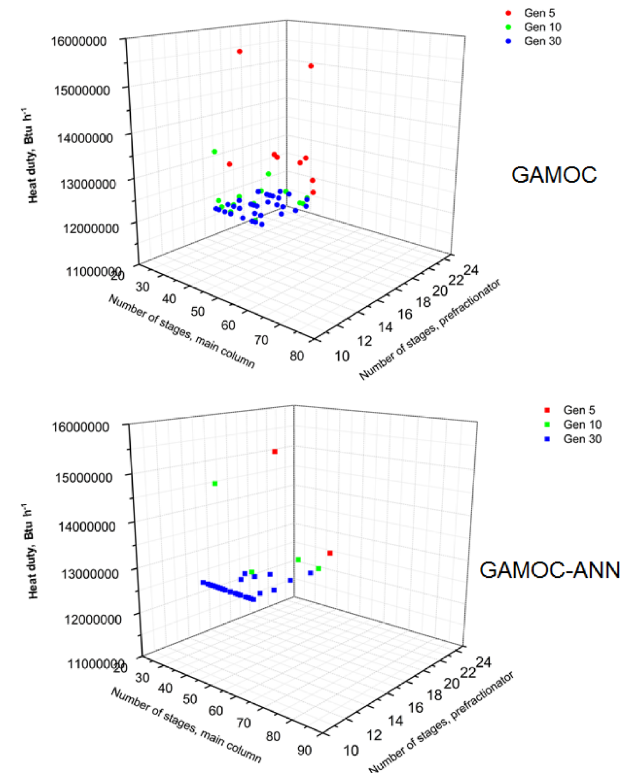


Fig. 5 – Evolution of the Pareto front of single dividing-wall column for mixture M2

According to Figure 6, the reduction in the number of evaluations of the complete functions is accompanied by a decreased reduction of the CPU time required when surrogate models are used. Figure 6 shows the CPU time required in the function of the number of generations. From this figure, we can observe the dramatic reduction in the computation time when surrogate models are used, 100,000 seconds, in comparison with GAMOC, 525,000 seconds. This represents a reduction of 80 % in the time required to reach the Pareto front. For mixture M2, a similar reduction in computation time required is observed when surrogate models are used. Therefore, with the use of neural networks, a major number of dividing-wall columns are evaluated in seconds, allowing us to explore a major area of the solution space in a short time, in comparison with a rigorous simulation.

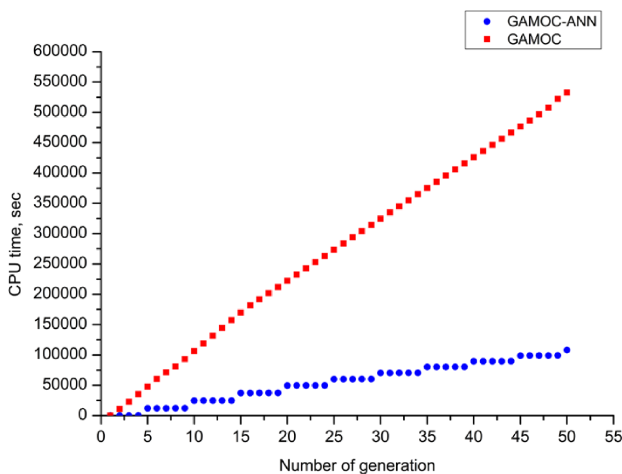


Fig. 6 – Required computation time as a function of the number of generations with GAMOC and GAMOC-ANN, for mixture M1

From the last figures, we can observe that the use of surrogate models allows decreasing the number of evaluations of the complete function, and, as consequence, the CPU required time. However, it is important to evaluate that these advantages do not impair the quality of the individuals that integrate the Pareto front; in other words, that the best individuals are found. In order to observe these characteristics, we present one graphic and one table.

The graphic shows the number of individuals with actual zero dominance as a function of the CPU time and the number of generations for mixture M1, Figure 7. It is worth mentioning that the individuals with actual zero dominance are those that integrate the Pareto front in a certain generation; in other words, the best individuals in that moment of the optimization procedure. From this figure, we observe that the number of individuals with actual zero dominance with the use of surrogate

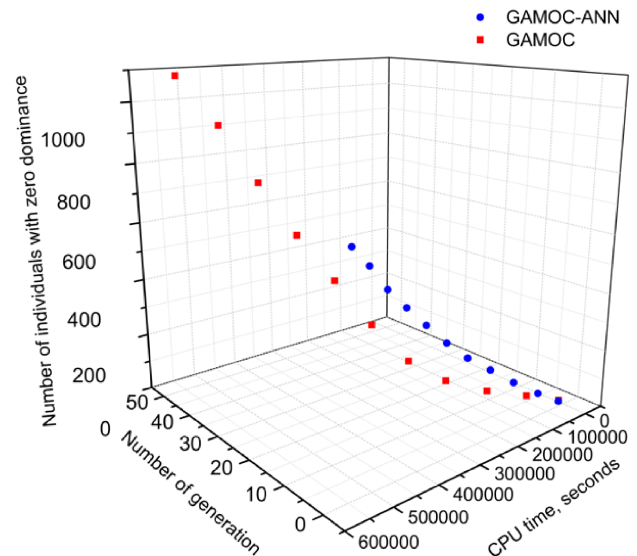


Fig. 7 – Number of individuals with zero dominance as function of the number of generations and the CPU time required, for mixture M1

models is slightly more than a third of the number obtained with GAMOC; nevertheless, the individuals with actual zero dominance of GAMOC-ANN were generated in just 20 % of the total time employed by GAMOC. In this sense, the trade-off between CPU time and number of individuals with actual zero dominance is good enough.

On the other hand, the quality of these individuals must be estimated, since it is not useful to have a lot of individuals if they pass from one generation to the other without any change. In order to observe this characteristic, we present in Table 2 the evolution of the individuals with actual zero dominance with respect to the minimum, maximum and the best trade-off between the objectives of interest. It is clear that, for both strategies, there is an improvement in the individuals; however, it is also clear that the improvements lead to better individuals with the use of surrogate models; is the consequence being that a major area of the solution space is being explored. It is important to remember that the time required to reach these better individuals is just 20 % of the total time employed when only complete models are used.

Another important aspect is the evaluation of the prediction of the artificial neural network. For this, we selected the individuals from several generations evaluated with the neural network, and which actual dominance was calculated as zero; we simulated these individuals in Aspen Plus, in order to obtain the values of the objectives and constraints resulting from using the complete functions. Only individuals with actual zero dominance were evaluated, since they are guiding the optimization procedure. Figures 8–10 show the difference between the

Table 2 – Evolution of the objective heat duty (Btu h^{-1}) in three individuals (minimum heat duty, minimum number of stages, and best trade-off) with zero dominance with GAMOC and GAMOC-ANN, for mixture M1

Generation	GAMOC-ANN			GAMOC		
	minimum heat duty	best trade-off	minimum stages	minimum heat duty	best trade-off	minimum stages
1	9 338 734	9 349 582	11 463 763	10 148 024	11 261 544	12 946 740
5	9 217 700	9 313 937	9 499 618	9 348 417	9 503 617	13 273 197
10	9 203 848	9 220 839	9 639 028	9 280 305	9 321 355	11 517 071
15	9 203 276	9 230 125	9 344 188	9 272 069	9 296 569	9 344 078
20	9 202 936	9 221 591	9 461 164	9 266 932	9 283 912	9 329 511
25	9 198 965	9 216 442	9 283 277	9 266 585	9 277 597	9 323 371
30	9 196 951	9 212 539	9 249 634	9 266 617	9 276 779	9 320 977
35	9 196 808	9 205 425	9 252 043	9 266 559	9 275 540	9 302 588
40	9 196 896	9 202 866	9 222 954	9 266 558	9 275 743	9 302 271
45	9 196 810	9 204 341	9 221 208	9 266 543	9 274 820	9 302 257
50	9 196 934	9 207 665	9 230 917	9 266 520	9 273 777	9 302 203

values of objectives and constraints obtained with complete and estimated functions, as the optimization proceeded. Generally, in all graphics, the difference is large at the beginning of the optimization procedure, but it is minimized as the process continues. In the case of heat duty, Figure 8, at the beginning of the optimization process, the maximum difference is around $60,000 \text{ Btu h}^{-1}$ (being the heat duty in the order of millions of Btu h^{-1}); at this point, the parameters of the neural network are obtained with numerous individuals that are far away from the optimal zone. The parameters of the neural network are refined as the optimization proceeds, and, at the end, the maximum difference is around 300 Btu h^{-1} . Since the optimal values are around 9 million Btu h^{-1} , the difference of 300 Btu h^{-1} is really small, 0.003% . On the other hand, Figure 9 shows the difference in the composition and recovery

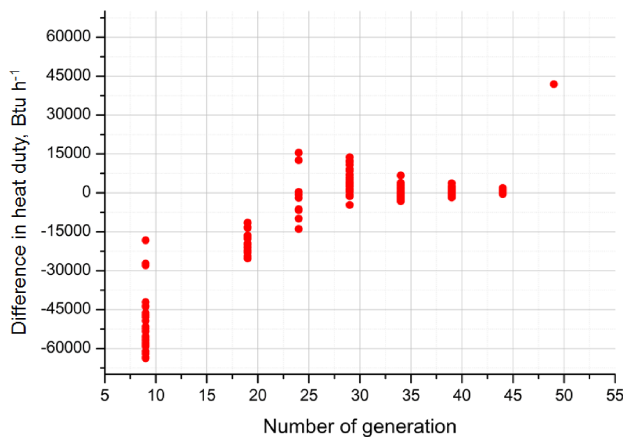


Fig. 8 – Deviation between the simulation of the complete model and the prediction of the neural network of the heat duty, mixture M1

of the light component with complete and estimated functions; in this figure, we observe that the differences are 0.005 at the beginning of the pro-

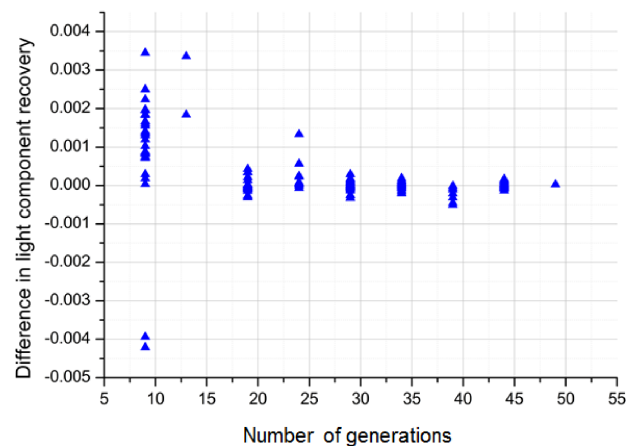
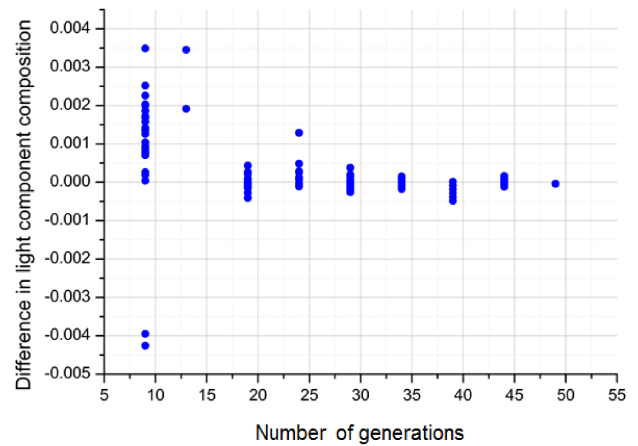


Fig. 9 – Deviation between the simulation of the complete model and the prediction of the neural network of the light component composition and recovery, mixture M1

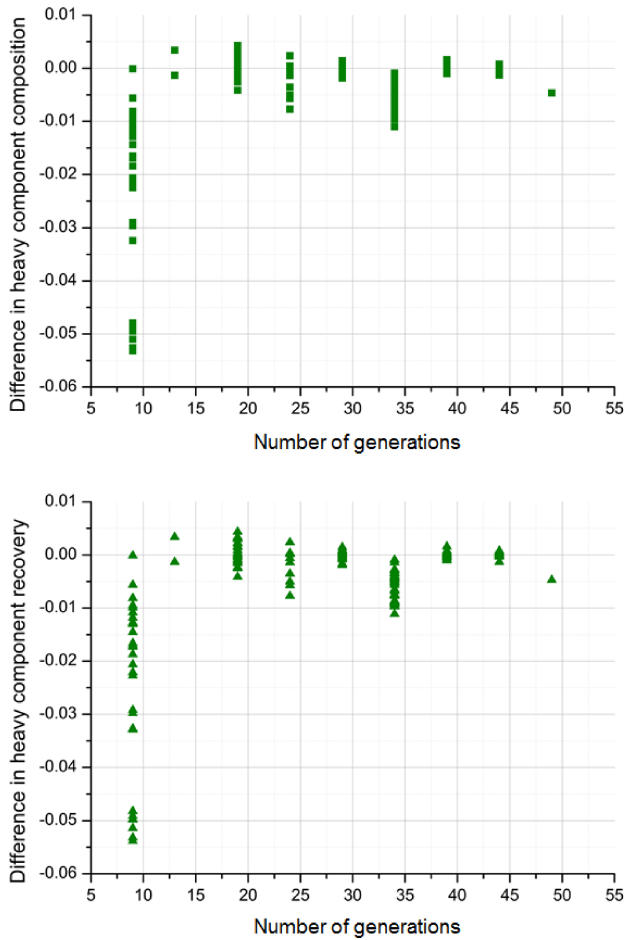


Fig. 10 – Deviation between the simulation of the complete model and the prediction of the neural network of the heavy component composition and recovery, mixture M1

cess, while at the end this value is $3.0 \cdot 10^{-5}$; considering that the values for composition goes from 0 to 1, that difference is really small. For the heavy component, Figure 10, the differences are larger at the beginning of the process, until 0.12 and 0.06, respectively; however, these differences are also minimal at the end of the process. In general, the predictions of the neural network were better as the optimization proceeded; this allows reaching the Pareto front very quickly, since the estimated objectives and constraints are very similar to those obtained with the complete functions.

Finally, among the Pareto fronts, we selected one optimal design in order to analyze the composition and temperature profiles. It is well known that the low thermodynamic efficiency of conventional distillation sequences is due to the remixing effect of the intermediate component for a ternary mixture. Figure 11 shows the liquid composition and temperature profiles on both sides of the dividing-wall for one optimal design of the Pareto front for mixture M1. In this design, the main column has 34 stages, where the side stream is located in stage

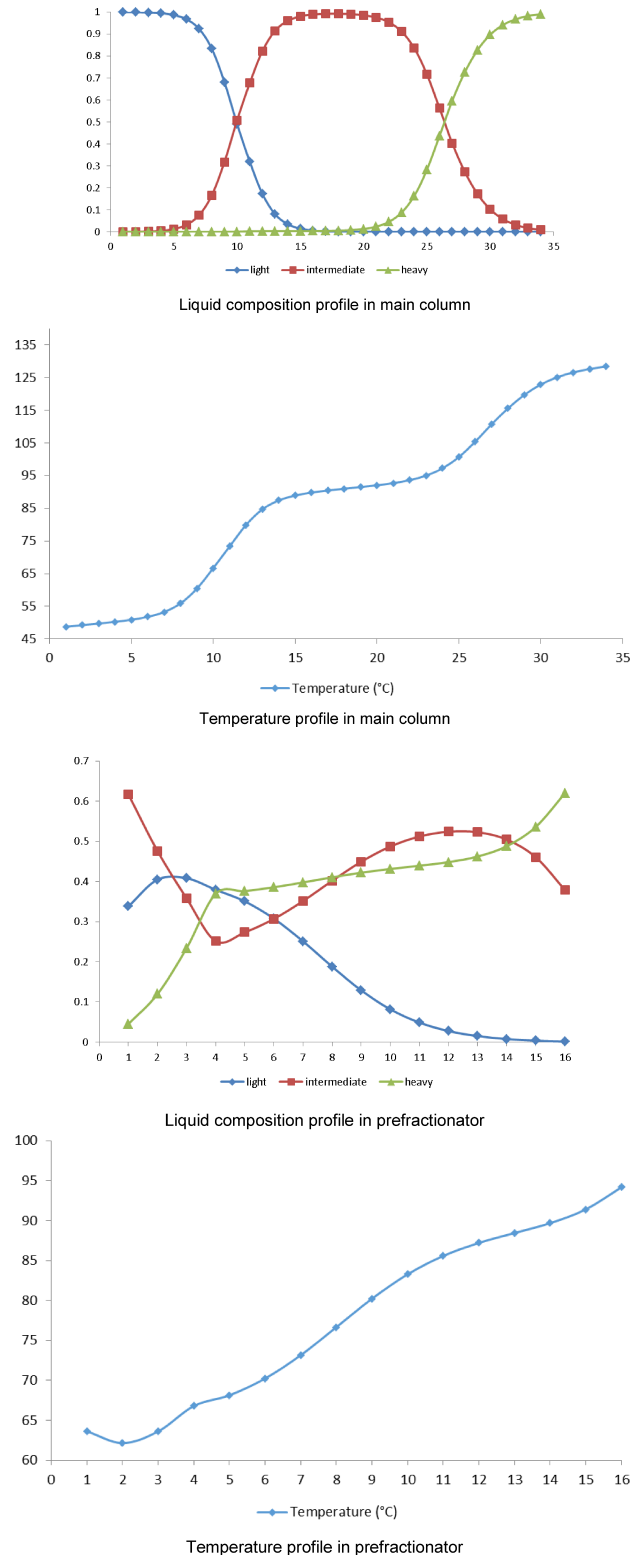


Fig. 11 – Liquid composition and temperature profiles on both sides of the dividing-wall column of one selected Pareto optimal for mixture M1

15 (counting from top to bottom); the prefractionator has 16 stages, being the feed stream located in stage 4 (counting from top to bottom). In the prefractionator, we observed that the temperature profile was really smooth, while the liquid composition

profile shows how a large part of the intermediate component along with the light component was distributed to the top of the prefractionator. In this case, this is the optimal distribution of the components for minimizing energy consumption and the number of stages. Moreover, in the main column, we observed that the liquid composition and temperature profiles were really smooth, indicating that the interconnection streams were optimally located, so there was no disturbance observed in the graphics. It is clear that both profiles were very soft, and no remixing effect was observed. This confirms that the reached designs represent optimal points.

Concluding remarks

The optimization of dividing-wall distillation columns by using surrogate models in a multiobjective stochastic algorithm has been presented. The proposed approach considers the use of artificial neural networks to generate approximate functions for objective and constraint functions. These approximate functions are several orders of magnitude less computationally expensive than the original ones. The results show a significant decrease in the number of evaluations of the original objective function required to reach the Pareto front; as a consequence, a significant decrease in the computation time was also observed.

The main weakness of this approach is the lack of sufficient data for the training of the neural network. Therefore, in cases of convergence problems in the simulator, the neural network cannot be trained in a reliable way, since there are insufficient data, and of course the strategy is not going to work properly.

It is important to mention that this approach can be used in any simple or multiobjective algorithm (stochastic or deterministic), with or without constraints, where the evaluation of the original objective function is computationally expensive. It is worth mentioning that, in the case of deterministic optimization strategies, a slight modification must be made. In spite of the selection of the neural network, it must be analyzed and validated for each case; results show that even the simpler neural network has a good performance, even in a very complex, mixed integer, highly nonlinear problem of optimal design.

ACKNOWLEDGEMENTS

The financial support provided by CONACyT (México) through project 239765 is gratefully acknowledged.

References

1. *Mezura-Montes, E.*, Uso de la técnica multiobjetivo NPGA para el manejo de restricciones en Algoritmos Genéticos. Maestría en Inteligencia Artificial, Thesis, Universidad Veracruzana (2001).
2. *Sarkar, D., Modak, J. M.*, Pareto-optimal solutions for multiobjective optimization of fed-batch bioreactors using non-dominated sorting genetic algorithm, *Chemical Engineering Science* **60**(2) (2005) 481. doi: <http://dx.doi.org/10.1016/j.ces.2004.07.130>
3. *Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.*, A fast elitist non-dominated sorting genetic algorithm for multiobjective-optimization: NSGA-II, KanGAL report 200001, Indian Institute of Technology, Kanpur, India (2000).
4. *Kasat, R. B., Gupta, S. K.*, Multi-objective optimization of an industrial fluidized-bed catalytic cracking unit (FCCU) using genetic algorithm (GA) with the jumping genes operator, *Computers and Chemical Engineering* **27**(12) (2003) 1785. doi: [http://dx.doi.org/10.1016/S0098-1354\(03\)00153-4](http://dx.doi.org/10.1016/S0098-1354(03)00153-4)
5. *Rasheed, K., Hirsh, H.*, Informed operators: Speeding up genetic-algorithm-based design optimization using reduced models, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)* (2000) 628.
6. *Doerr, B., Hebbinghaus, N., Neumann, F.*, Speeding up Evolutionary Algorithms through Asymmetric Mutation Operators, *Evolutionary Computation* **15**(4) (2007) 401. doi: <http://dx.doi.org/10.1162/evco.2007.15.4.401>
7. *Tikidji-Hamburyan, R. A.*, Genetic algorithm modification to speed up parameter fitting for a multicompartment neuron model, *Proceedings of Seventeenth Annual Computational Neuroscience Meeting* **9**(Suppl 1) (2008) P90. doi: <http://dx.doi.org/10.1186/1471-2202-9-s1-p90>
8. *Poloni, C., Giurgevich, A., Onesti, L., Pedirola, V.*, Hybridization of a multi-objective genetic algorithm, a neural network and a classical optimizer for complex design problem in fluid dynamics, *Computers Methods in Applied Mechanics and Engineering* **186** (2000) 403. doi: [http://dx.doi.org/10.1016/S0045-7825\(99\)00394-1](http://dx.doi.org/10.1016/S0045-7825(99)00394-1)
9. *Nain, P. K. S., Deb, K.*, A computationally effective multi-objective search and optimization technique using coarse-to-fine grain modeling, *Kangal Report No. 2002005* (2002).
10. *Gaspar-Cunha, A., Vieira, A.*, A multiobjective evolutionary algorithm using neural networks to approximate fitness evaluations, *International Journal of Computers, Systems and Signals* **6**(1) (2005) 18.
11. *Gaspar-Cunha, A., Covas, J. A.*, RPSGAe – A multiobjective genetic algorithm with Elitism: Applications to Polymer Extrusion, *Lecture Notes in Economics and Mathematical Systems*, Springer (2004).
12. *Zhou, Z., Ong, Y. S., Nair, P. B., Keane, A. J., Lum, K. Y.*, Combining global and local surrogate models to accelerate evolutionary optimization, *IEEE Transactions on Systems, man and cybernetics – Part C: Applications and Reviews* **37**(1) (2007) 66. doi: <http://dx.doi.org/10.1109/TSMCC.2005.855506>
13. *Sreekanth, J., Datta, B.*, Multi-objective management of saltwater intrusion in coastal aquifers using genetic programming and modular neural network based surrogate models, *Journal of Hydrology* **393** (2010) 245. doi: <http://dx.doi.org/10.1016/j.jhydrol.2010.08.023>
14. *Kim, H. J., Chang, K. S.*, A method of model validation for chaotic chemical reaction systems based on neural network, *Korean Journal of Chemical Engineering* **18**(5) (2001) 623. doi: <http://dx.doi.org/10.1007/BF02706377>

15. *Gomes, M. V. C., Bogle, I. D. L., Biscaia Jr., E. C., Odloak, D.*, Using kriging models for real-time process optimisation, *Computer Aided Chemical Engineering* **25** (2008) 361.
doi: [http://dx.doi.org/10.1016/S1570-7946\(08\)80065-X](http://dx.doi.org/10.1016/S1570-7946(08)80065-X)
16. *Ho, T. C.*, Kinetic modeling of large-scale reaction systems, *Catalysis Reviews: Science and Engineering* **50**(3) (2008) 287.
doi: <http://dx.doi.org/10.1080/01614940802019425>
17. *Henao, C. A., Maravelias, C. T.*, Surrogate-based superstructure optimization framework, *AIChE Journal* **57**(5) (2010) 1216.
doi: <http://dx.doi.org/10.1002/aic.12341>
18. *Wenxiang, L., Ying, Z., Dexian, H., Yongheng, J., Yihui, J.*, A New Strategy of Integrated Control and On-line Optimization on High-purity Distillation Process, *Chinese Journal of Chemical Engineering* **18**(1) (2010) 66.
doi: [http://dx.doi.org/10.1016/S1004-9541\(08\)60325-0](http://dx.doi.org/10.1016/S1004-9541(08)60325-0)
19. *Bojarski, A. D., Pérez-Fortes, M., Nougues, J. M., Puigjaner, L.*, Modelling Superstructure for Conceptual Design of Syngas Generation and Treatment, *Syngas from waste, Green energy and technology series*, Springer, 2011, pp 169-199.
20. *Biegler, L. T., Lang, Y.-D.*, Multi-scale optimization for advanced energy process, *Proceedings of the 11th International Symposium on Process Systems Engineering*, Elsevier, 2012, pp 51-60.
doi: <http://dx.doi.org/10.1016/B978-0-444-59507-2.50007-X>
21. *Fahmi, I., Cremaschi, S.*, Process synthesis of biodiesel production plant using artificial neural networks as the surrogate models, *Computers and Chemical Engineering* **46** (2012) 105.
doi: <http://dx.doi.org/10.1016/j.compchemeng.2012.06.006>
22. *Caballero, J. A., Navarro, M. A., Ruiz-Femenia, R., Grossmann, I. E.*, Integration of different models in the design of chemical processes: Application to the design of a power plant, *Applied Energy* **124** (2014) 256.
doi: <http://dx.doi.org/10.1016/j.apenergy.2014.03.018>
23. *Stankiewicz, A. I., Moulijn, J. A.*, Process Intensification: Transforming Chemical Engineering, *Chemical Engineering Progress*, January (2000) 22.
24. *Gutiérrez-Antonio, C., Briones-Ramírez, A.*, Pareto front of ideal Petlyuk sequences using a multiobjective genetic algorithm with constraints, *Computers and Chemical Engineering* **33**(2) (2009) 454.
doi: <http://dx.doi.org/10.1016/j.compchemeng.2008.11.004>
25. *Whitley, D.*, Genetic Algorithms and Neural Networks, Chapter 11, *Genetic Algorithms in Engineering and Computer Science*, 1995.
26. *Zhang, Byoung-Tak, Mühlenbein, H.*, Evolving optimal Neural Networks Using Genetic Algorithms with Occam's Razor, *Complex Systems* **7**(3) (1993) 199.
27. *Man, K. F., Tang, K. S., Kwong, S.*, Genetic Algorithms: Concepts and Applications, *IEEE Transactions on Industrial Electronics* **43**(5) (1996) 519.
doi: <http://dx.doi.org/10.1109/41.538609>
28. *Agatonovic-Kustrin, S., Beresford, R.*, Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research, Review, *Journal of Pharmaceutical and Biomedical Analysis* **22** (2000) 717.
doi: [http://dx.doi.org/10.1016/S0731-7085\(99\)00272-1](http://dx.doi.org/10.1016/S0731-7085(99)00272-1)
29. *Kurková, V.*, Kolmogorov's theorem and multilayer neural networks, *Neural Networks* **5**(3) (1992) 501.
doi: [http://dx.doi.org/10.1016/0893-6080\(92\)90012-8](http://dx.doi.org/10.1016/0893-6080(92)90012-8)
30. *Coello-Coello, C. A.*, Constraint-Handling using an Evolutionary Multiobjective Optimization Technique, *Civil Engineering and Environmental Systems* **17** (2000) 319.
doi: <http://dx.doi.org/10.1080/02630250008970288>
31. *Schultz, M. A., Stewart, D. G., Harris, J. M., Rosenblum, S. P., Shakur, M. S., O'Brien, D. E.*, Reduce costs with dividing-wall columns, *Chemical Engineering Progress* **98** (2002) 64.
32. *Dejanović, I., Matijašević, Lj., Olujić, Ž.*, Dividing wall column – A breakthrough towards sustainable distilling, *Chemical Engineering and Processing: Process Intensification* **49**(6) (2010) 559.
doi: <http://dx.doi.org/10.1016/j.ccep.2010.04.001>
33. *Gutiérrez-Antonio, C., Hernández, S., Gómez-Castro, F. I., Segovia-Hernández, J. G., Campos-Vargas, J. O., Briones-Ramírez, A.*, Optimal design and control of trains of dividing wall columns for the separation of petrochemical mixtures, *Computer Aided Chemical Engineering* **30** (2012) 742.
doi: <http://dx.doi.org/10.1016/B978-0-444-59520-1.50007-5>
34. *Rodríguez-Ángeles, M. A., Gómez-Castro, F. I., Segovia-Hernández, J. G., Gutiérrez-Antonio, C., Briones-Ramírez, A., Uribe-Ramírez, A. R.*, Performance of Dividing Wall Columns for the Separation of Light Hydrocarbons, *Advances in Petrochemical Engineering II: Petrochemical* **2** (2013) 43.
35. *Tavan, Y., Shahhosseinib, S., Hosseini, S. H.*, Design and simulation of ethane recovery process in an extractive dividing wall column, *Journal of Cleaner Production* **72** (2014) 222.
doi: <http://dx.doi.org/10.1016/j.jclepro.2014.03.015>
36. *Briones-Ramírez, A., Gutiérrez-Antonio, C.*, Dividing wall distillation columns for separation of azeotropic mixtures: feasibility procedure and rigorous optimization, *Computer Aided Chemical Engineering* **26** (2009) 555.
doi: [http://dx.doi.org/10.1016/S1570-7946\(09\)70093-8](http://dx.doi.org/10.1016/S1570-7946(09)70093-8)
37. *Kiss, A. A., Suszwalak, D. J. P. C.*, Enhanced bioethanol dehydration by extractive and azeotropic distillation in dividing-wall columns, *Separation and Purification Technology* **86** (2012) 70.
doi: <http://dx.doi.org/10.1016/j.seppur.2011.10.022>
38. *Hernández, S., Segovia-Hernández, J. G., Santamaría-Rivera, M. P., Hernández-Escoto, H., Gutiérrez-Antonio, C., Briones-Ramírez, A., Maya-Yescas, R.*, Bioethanol: Production, Benefits and Economics, Jason B. Erbau Editor. Nova Science Publishers, Inc., 2009, pp 119-134.
39. *Sanchez-Ramírez, E., Quiroz-Ramírez, J. J., Segovia-Hernández, J. J., Hernández, S., Bonilla-Petriciolet, A.*, Process Alternatives for Biobutanol Purification: Design and Optimization, *Industrial and Engineering Chemistry Research* **54** (2015) 351.
doi: <http://dx.doi.org/10.1021/ie503975g>
40. *Gutiérrez-Antonio, C., Gómez-Castro, F. I., Hernández, S., Briones-Ramírez, A.*, Intensification of a hydrotreating process to produce biojet fuel using thermally coupled distillation, *Chemical Engineering and Processing: Process Intensification* **88** (2015) 29.
doi: <http://dx.doi.org/10.1016/j.ccep.2014.12.002>
41. *Miranda-Galindo, E. Y., Segovia-Hernández, J. G., Hernández, S., Gutiérrez-Antonio, C., Briones-Ramírez, A.*, Reactive Thermally Coupled Distillation Sequences: Pareto Front, *Industrial and Engineering Chemistry Research* **50** (2011) 926.
doi: <http://dx.doi.org/10.1021/ie101290t>
42. *Wanga, S.-J., Chenb, W.-Y., Chang, W.-T., Hu, C.-C., Cheng, S. H.*, Optimal design of mixed acid esterification and isopropanol dehydration systems via incorporation of

- dividing-wall columns, *Chemical Engineering and Processing: Process Intensification* **85** (2014) 108.
doi: <http://dx.doi.org/10.1016/j.cep.2014.08.011>
43. *Hernández, S., Sandoval-Vergara, R., Barroso-Muñoz, F. O., Murrieta-Dueñas, R., Hernández-Escoto, H., Segovia-Hernández, J. G., Rico-Ramírez, V.*, Reactive dividing wall distillation columns: Simulation and implementation in a pilot plant, *Chemical Engineering and Processing: Process Intensification* **48**(1) (2009) 250.
doi: <http://dx.doi.org/10.1016/j.cep.2008.03.015>
44. *Rong, B. G.*, Synthesis of dividing-wall columns (DWC) for multicomponent distillations—A systematic approach, *Chemical Engineering Research and Design* **89**(8) (2011) 1281.
doi: <http://dx.doi.org/10.1016/j.cherd.2011.03.014>
45. *Gómez-Castro, F. I., Rodríguez-Ángeles, M. A., Segovia-Hernández, J. G., Gutiérrez-Antonio, Briones-Ramírez, A.*, Optimal Designs of Multiple Dividing Wall Columns, *Chemical Engineering Technology* **34**(12) (2011) 2051.
doi: <http://dx.doi.org/10.1002/ceat.201100176>
46. *Premkumar, R., Rangaiah, G. P.*, Retrofitting conventional column systems to dividing-Wall Columns, *Chemical Engineering Research and Design* **87**(1) (2009) 47.
doi: <http://dx.doi.org/10.1016/j.cherd.2008.06.013>
47. *Asprion, N., Kaibel, G.*, Dividing wall columns: Fundamentals and recent advances, *Chemical Engineering and Processing: Process Intensification* **49**(2) (2010) 139.
doi: <http://dx.doi.org/10.1016/j.cep.2010.01.013>
48. *Vazquez-Castillo, J. A., Venegas-Sánchez, J. A., Segovia-Hernández, J. G., Hernández-Escoto, H., Hernández, S., Gutiérrez-Antonio, C., Briones-Ramírez, A.*, Design and optimization, using genetic algorithms, of intensified distillation systems for a class of quaternary mixtures, *Computers and Chemical Engineering* **33**(11) (2009) 1841.
doi: <http://dx.doi.org/10.1016/j.compchemeng.2009.04.011>
49. *Kim, S. B., Linninger, A. E.*, Optimization of Complex Column Networks with Hybrid Genetic Algorithm, *Computer Aided Chemical Engineering* **31** (2012) 1597.
doi: <http://dx.doi.org/10.1016/B978-0-444-59506-5.50150-4>
50. *Gutiérrez-Antonio, C., Briones-Ramírez, A., Jiménez-Gutiérrez, A.*, Optimization of Petlyuk sequences using a multi objective genetic algorithm with constraints, *Computers and Chemical Engineering* **35**(2) (2011) 236.
doi: <http://dx.doi.org/10.1016/j.compchemeng.2010.10.007>
51. *Ge, X., Yuan, X., Ao, Ch., Yu, K.-K.*, Simulation based approach to optimal design of dividing wall column using random search method, *Computers and Chemical Engineering* **68** (2014) 38.
doi: <http://dx.doi.org/10.1016/j.compchemeng.2014.05.001>
52. *Torres-Ortega, C. E., Errico, M., Rong, B.-G.*, Design and optimization of modified non-sharp column configurations for quaternary distillations, *Computers and Chemical Engineering* **74** (2015) 15.
doi: <http://dx.doi.org/10.1016/j.compchemeng.2014.12.006>
53. *Errico, M., Pirellas, P., Torres-Ortega, C. E., Rong, B.-G., Segovia-Hernández, J. G.*, A combined method for the design and optimization of intensified distillation systems, *Chemical Engineering and Processing: Process Intensification* **85** (2014) 69.
doi: <http://dx.doi.org/10.1016/j.cep.2014.08.005>
54. *Kiss, A. A., Segovia-Hernández, J. G., Bildea, C. S., Miranda-Galindo, E. Y., Hernández, S.*, Innovative biodiesel production in a reactive dividing-wall column, *Computer Aided Chemical Engineering* **30** (2012) 522.
doi: <http://dx.doi.org/10.1016/B978-0-444-59519-5.50105-2>
55. *Yuan, X.-G., An, W.-Z.*, Synthesis of fully thermally coupled distillation columns for multicomponent separation via stochastic optimization, *Computer Aided Chemical Engineering* **15** (2003) 1123.
doi: [http://dx.doi.org/10.1016/S1570-7946\(03\)80459-5](http://dx.doi.org/10.1016/S1570-7946(03)80459-5)
56. *Long, N. V. C., Kon, Y., Lee, M.*, Design and optimization of thermally coupled distillation schemes for the trichlorosilane purification process, *Applied Thermal Engineering* **59**(1-2) (2013) 200.
doi: <http://dx.doi.org/10.1016/j.applthermaleng.2013.05.035>
57. *Amminudin, K. A., Smith, R., Thong, D. Y.-C., Towler, G. P.*, Design and Optimization of Fully Thermally Coupled Distillation Columns: Part 1: Preliminary Design and Optimization Methodology, *Chemical Engineering Research and Design* **79**(7) (2001) 701.
doi: <http://dx.doi.org/10.1205/026387601753192028>
58. *Ignat, R. M., Kiss, A. A.*, Optimal design, dynamics and control of a reactive DWC for biodiesel production, *Chemical Engineering Research and Design* **91**(9) (2013) 1760.
doi: <http://dx.doi.org/10.1016/j.cherd.2013.02.009>
59. *Caballero, J. A., Grossmann, I. E.*, Optimal synthesis of thermally coupled distillation sequences using a novel MILP approach, *Computers and Chemical Engineering* **61** (2014) 118.
doi: <http://dx.doi.org/10.1016/j.compchemeng.2013.10.015>
60. *Grossmann, I. E., Aguirre, P. A., Bartfeld, M.*, Optimal synthesis of complex distillation columns using rigorous models, *Computers and Chemical Engineering* **29**(6) (2005) 1203.
doi: <http://dx.doi.org/10.1016/j.compchemeng.2005.02.030>
61. *Kiss, A. A., Ignat, R. M., Flores Landaeta, S. J., de Haan, A. B.*, Intensified process for aromatics separation powered by Kaibel and dividing-wall columns, *Chemical Engineering and Processing: Process Intensification* **67** (2013) 39.
doi: <http://dx.doi.org/10.1016/j.cep.2012.06.010>