I&EC research
Industrial & Engineering Chemistry Research

Article

pubs.acs.org/IECR

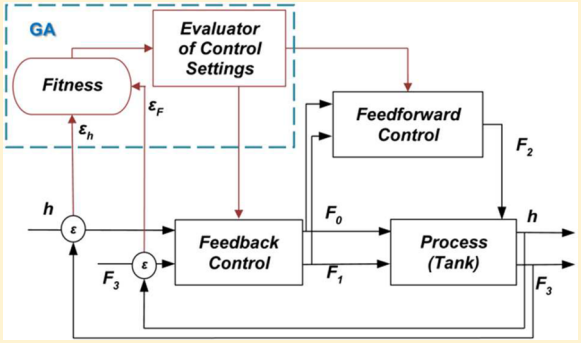
An Evolutionary Algorithm Applied to Inventory Control for Natural Gasoline

Paola Patricia Oteiza[†,‡] and Nélida Beatriz Brignole*[,†,‡]

[†]Laboratorio de Investigación y Desarrollo en Computación Científica (LIDECC)-Departamento de Ciencias e Ingeniería de la Computación (DCIC), Universidad Nacional del Sur (UNS), Bahía Blanca, Argentina

[‡]Planta Piloto de Ingeniería Química (PLAPIQUI), Universidad Nacional del Sur-CONICET, Bahía Blanca, Argentina

ABSTRACT: A metaheuristic technique for controller design is explored, and its performance is assessed for a real-world case. An evolutionary algorithm for self-tuning systems is proposed to tune the optimal control settings in order to achieve the best overall performance to control accurately the volume of natural gasoline in a storage tank. For high-level optimization, a flexible iterative technique based on Genetic Algorithms is presented. The algorithm is stable, and its architecture is simple, thus becoming easy to implement. Not only heuristics that produce superior solutions are taken into account, but also the computational speed is contemplated as a key factor. The experimental evaluation yields satisfactory fitness values, and relatively little computational effort is required.



1. INTRODUCTION

A wide variety of adaptive techniques are frequently used in controller design with a robust performance. Nevertheless, the systems with a fixed controller invariant in time turn out to be more insensitive to disturbances.[1] Therefore, some efforts have lately been made on the formulation of PI (Proportional−Integral) or PID (Proportional−Integral−Derivative) controller tuning rules. For a wide range of processes, Shamsuzzoha has derived simple correlations to yield PI/PID controller settings with acceptable performance and robustness.[2] In particular, Cho et al. have proposed PID controller tuning rules for unstable processes.[3] Also, Li et al. have implemented a self-tuning figure that automatically tunes the settings for local adjustment of water quality.[4]

The implementation of self-tuning control systems is still a big challenge for practical control engineers, and the subject remains open for research. Moreover, it is worthwhile making efforts in close connection with Evolutionary Computation (EC) dedicated to this problem because EC is an established field for optimization whenever it is necessary to perform an exhaustive search in relatively short computing times. Evolutionary algorithms (EAs) are suitable to manage a wide range of computationally intractable problems since they constitute a simple, flexible approach. Though they do not always guarantee that the exact optimal solution will be found in a single run, they can often obtain satisfactory solutions within acceptable computational times.[5] If the entire domain of definition of the objective function is to be explored, there are several metaheuristics that can be employed in practice to obtain a solution, including genetic algorithms (GAs), evolution strategies, differential evolution, particle swarm optimization, and others.[6]

As to self-tuning systems, some metaheuristics have already been used. Oh et al. proposed a genetic-based self-tuning speed controller for the high-performance drives of induction motors.[7] In contrast, Liu and Hsu designed a self-tuning PI controller by means of Particle Swarm Optimization.[8,9]

Only regarding controller design with metaheuristic tools, Viswanathan et al.[10] successfully addressed the closed-loop identification problem of two-input two-output processes employing a GA to locate reliably the global minimum of the least-squares problem.

Almost simultaneously, Deb et al.[11] introduced the concepts of Non-Dominated Sorting GA-II (NSGA-II), which has proved to be a successful approach to solve multiobjective problems. This algorithm has been employed in various chemical processes, which are listed in Behroozsarand and Shafiei.[12] In particular, these authors tuned PID controllers in the optimal control of an amine plant by minimizing two objective functions related to the overshoot and the Integral Absolute Error through the (NSGA-II). They have stated that good control profiles could be achieved, resulting in an acceptable compromise between conflicting objectives. Recently, Ayala and dos Santos Coelho[13] presented the design and the tuning of two PID controllers in a robotic manipulator of two degrees of freedom, also using the NSGA-II approach. Besides, there are other innovative efforts to specifically improve GA tactics for PID tuning. Chang[14] proposed a modified crossover formula in GAs and used this method to

Received: May 19, 2016
Revised: December 2, 2016
Accepted: December 4, 2016
Published: December 5, 2016

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062−13073

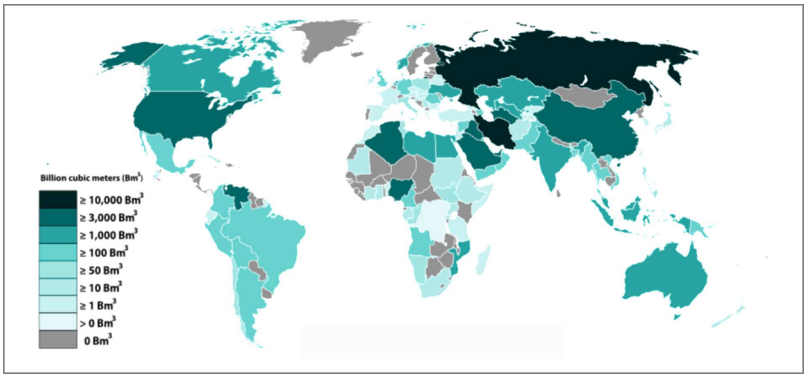


Figure 1. Salient distribution of natural gas proven reserves in 2014.

determine PID controller gains for multivariable processes. Moreover, Meng and Song[15] presented a fast GA, exhibiting improvements about population, selection, crossover, and mutation in comparison with simple GAs. On the other hand, Chan et al. proposed a gene manipulation, which is a multiobjective GA to optimize the placement of active devices and sensors in frame structures to reduce active control cost and increase the structural control strategy's effectiveness.[16] In turn, Neath et al. proposed a GA to tune the settings of a PID controller for a bidirectional inductive power transfer system, taking various objectives functions (fitness) into consideration, to achieve a controller with optimal performance.[17] Recently, Tochampa et al. presented a model-based optimization involving a GA for the optimal control of the feeding profile in fed-batch production of xylitol, being the chromosomes defined with feed rate elements.[18]

In this paper, we have addressed the inventory control of natural gasoline extracted directly from the gas fields and transported by pipeline networks to a storage tank. It is assumed that the natural gasoline should next be sent with a steady flow to a processing plant. The system under adaptive control is the storage tank, where liquid hydrocarbons are collected by means of pipelines coming from various sources with their own inherent dynamics, which is specially contemplated in our design.

It is interesting to note that enterprise-wide optimization is progressively moving from a conventional steady-state tactics toward a dynamic real-time approach.[19] In view of both this trend and the recent advances concerning EC, one of our main purposes is to extend the nature-inspired framework to solve the following challenging self-tuning problem: the permanent regulation of nonlinear disturbances originated in the gas fields.

Section 2 enhances the present-day importance of natural gasoline by giving some general information about its uses and the reserves, while Section 3 describes the real-world problem under study. Next, Section 4 outlines the optimization methodology in general terms. The control strategy is presented in Section 5, followed by the analysis and discussion of the results in Section 6. Finally, some conclusions are presented in Section 7.

2. USES AND RESERVES OF NATURAL GASOLINE

At atmospheric temperatures and pressures, natural gasoline is a liquid, which is a byproduct of natural gas, and consists predominantly of pentane and upper hydrocarbons. Therefore, it can be transported and stored without pressurized containment. Natural gasoline can be obtained by primary separation, which usually takes place at the reservoirs of natural gas in basins around the world.[20] Then, natural gasoline well productivity is directly related to natural gas production and reserves. The map in Figure 1 illustrates the countries where there are natural gas proven reserves.[21] This distribution shows how promising it might be to exploit natural gasoline in some reservoirs. In particular, the production of natural gasoline has increased meaningfully in the United States since 2007, while exports of this hydrocarbon to Canada have surged.[22] Moreover, according to the U.S. Energy Information Administration, the latest surveys revealed that crude oil and lease condensate reserves have sustainably increased in the United States.[23] More precisely, at the end of 2015, these reserves totalized 2 billion bbl more than the amount reported a year before.[24]

Being more specific particularly about the Southern hemisphere, Argentina looks like a favorable location, with an annual production of 1739934 tones of natural gasoline in 2013.[25]

According to the Secretaría de Energía, during 2014, the Province of Santa Cruz (Argentina) yielded 9% gas production in the country.[26] Its main basins are located in Cuenca Austral and Cuenca Golfo San Jorge. In 2014, most of the natural gasoline production came from the latter. Table 1 summarizes the amounts of the secondary production of natural gasoline provided by the main wells located in the Province of Santa Cruz.

Table 1. Production of Natural Gasoline During 2014 (Cuenca Golfo San Jorge, Argentina)

fields	production [m^3]
Cañadón Yatel	16,224
El Huemul	14,300
Sur Piedra Clavada	8482
Cañadón León	7739
Cañadón Seco	7104
Estancia la Mariposa	4382
El Cóndor	3432
Pico Trucado	1978
Bayo	954

Nowadays, the natural gasoline remains unexploited in Argentina. Instead, it is mixed and transported with the crude or else accumulated in the fields. As a petrochemical raw material, natural gasoline can be employed to obtain ethylene-propylene via thermal cracking. Moreover, natural gasoline is a

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062−13073

convenient substitute for ethylene production in Argentina, as shown in Cañete et al.[27]

A petrochemical project can be considered economically attractive provided large scale plants are installed. For the transport of natural gasoline, the distance between the reservoirs and the petrochemical plants is significant. Pipeline networks, which are generally convenient for long distances, consist of three main parts: pipelines, pump stations, and storage tanks. In view of this scenario, transport should be optimized, and therefore, it is convenient to take into account the dynamics of the fields in the control system to have enough feedstock always available.

In contrast with other means of transport for the delivery of oil derivatives to distant destinations, pipeline transportation is the one that yields lower variable costs, although building costs of a pipeline network are high. Besides, once it is operative, it can function all the time, and the network only employs energy in order to move the product itself but not the containers.[28]

3. PROBLEM STATEMENT

The case under study (Figure 2) belongs to Cuenca Golfo San Jorge and comprises a concentrating node (the storage tank),

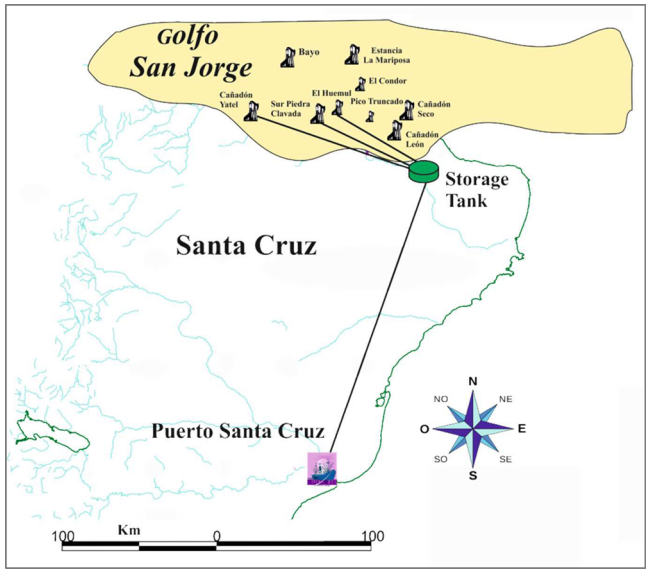


Figure 2. Simplified scheme of the transport network.

its inputs coming from the gas fields known as Cañadón Yatel (F_0), El Huemul (F_1), and Sur Piedra Clavada (F_2), and its output going to a processing plant.

In conventional reservoirs, where oil and gas can flow freely, economic risk is mainly driven by the uncertainties about the size and presence of hydrocarbon accumulations. Though this geological risk prevails early in a field's exploration and evaluation, it ameliorates as initial wells are drilled for the predictions about production can be made with certitude.[29]

In particular, the case illustrated in Figure 2 is located in Argentina, where natural gasoline is extracted from natural gas fields. Therefore, facing this problem is nowadays of wide interest taking into account that the natural gas world production is expected to reach its peak level around 2030.[30] Moreover, Maggio and Cacciola[31] have estimated that the peak for natural gas worldwide will occur between 2024 and 2046. The field's production profile $P_F(t)$ can be expressed mathematically as a function of time t (eq 1), where Y_F is the year when the field commences, and the field takes t_F years to reach the maximum production plateau F_p. At t_r, the remaining recoverable resources in the field reach the value Q_r, and after this time, the production begins to decline exponentially.[32]

$$P_F(t) = \begin{cases} 0 & \text{if } t < Y_F \\ \frac{F_p}{t_F}(t - Y_F) & \text{if } Y_F \le t < Y_F + t_F \\ F_p & \text{if } Y_F + t_F \le t < t_r \\ F_p e^{(-F_p(1-0.01)/Q_r(t-t_r))} & \text{if } t_r \le t < t_r - \frac{\log(0.01)Q_r}{F_p(1-0.01)} \\ 0 & \text{if } t > t_r - \frac{\log(0.01)Q_r}{F_p(1-0.01)} \end{cases} \quad (1)$$

As to investments in infrastructure, companies are sometimes hindered by the uncertain behavior of fields. A proper control strategy for managing these resources is proposed in this paper. Several control structures were analyzed, and the choice determines how well the system can be controlled to guarantee a stable output to satisfy external demands. The choice of an adaptive online control policy determines the best possible performance to be achieved thanks to well-tuned controllers. The fine-tuning procedure by means of GA can be repeated at regular time intervals so that the gasoline inventory and its outlet flow are effectively managed. By means of this approach, satisfactory tracking performance in a closed loop can be achieved.

4. RELATED BACKGROUND

Multiobjective optimization refers to the process of finding feasible solutions to a problem by trading off the equally optimal values of several functions subjected to a set of constraints. A multiobjective optimization problem can formally be defined as is given in eq 2.

$$\begin{aligned} &\min \boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{F}(f_1(\boldsymbol{x}), ..., f_M(\boldsymbol{x})) \\ &\text{Subjected to:} \\ &c_1(\boldsymbol{x}), ..., c_C(\boldsymbol{x}) \le 0 \\ &d_1(\boldsymbol{x}), ..., d_D(\boldsymbol{x}) = 0 \\ &\text{with } \boldsymbol{x} \in D \end{aligned} \quad (2)$$

where D is the decision space, x is the real N-vector whose elements are the N decision variables, $F(x)$ is the real-valued multiobjective optimization function, and $f_1(x),...f_M(x)$ are the M-associated objective functions.

Here, $F(x)$ can be established as the weighed sum of the objective functions (eq 3), where the weight w_i for the objective function f_i can be established.

$$\boldsymbol{F}(\boldsymbol{x}) = \sum_{i=1}^{M} w_i f_i(\boldsymbol{x}) \quad (3)$$

Finally, $c_1(x),...,c_C(x) \le 0$ and $d_1(x),...,d_D(x) = 0$ express the constraints imposed on the values of x.

The GA shown in Figure 3 is the EA that is adopted to carry out the minimization. Therefore, the function $F(x)$ can be named the fitness function. In particular, this algorithm is called a Multiobjective Evolutionary Algorithm (MOEA)[33] because a multiobjective function is employed for step 8. The execution

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062−13073

Genetic Algorithm (GA)

1. g_0:=0; // The first generation g_0
2. *Initialize* $\mathcal{P}(g_0)$; // A random population $\mathcal{P}$ is generated
3. **Do until** termination criterion is met: // Continue the iterations because the population set needs updating
    4. $g = g + 1$; // The next generation
    5. $\mathcal{P}(g)$:=*choose* $\mathcal{P}(g-1)$; // The best solution is chosen
    6. *Cross* $\mathcal{P}(g)$ *over*; // The candidates are crossed over
    7. *Mutate* $\mathcal{P}(g)$; // The offsprings are mutated
    8. *Evaluate* $\mathcal{P}(g)$ // The fitness $\boldsymbol{F}(\boldsymbol{x})$ of the individuals $\boldsymbol{x}$ is evaluated
9. **End do**

Figure 3. GA pseudocode.

of the GA terminates at the moment when any of the following conditions takes place:

Condition 1: The predefined maximum number of generations N_{max} has been reached.

Condition 2: The method keeps yielding a feasible solution that is unlikely to be improved. This is detected when the best values of the multiobjective optimization function $\hat{F}_g$ do not change significantly as generations go by. The convergence test checks for a sufficient relative decrease in $\hat{F}_g$ between one iteration/generation and the next by tracking the evolution of the relative error ϵ_g^F (eq 4) as generations move past. If $\epsilon_g^F \leq$ TOL, the iterations stop.

$$\epsilon_g^F = \frac{(|\widehat{\boldsymbol{F}_g} - \widehat{\boldsymbol{F}_{g-1}}|)}{|\widehat{\boldsymbol{F}_{g-1}}|} \tag{4}$$

A GA can be classified as a population-based evolutionary optimization method whose search policy was inspired in Darwin's principle of biological evolution.[34] Since GA is a search technique, it has to be limited to exploring a reasonable region of variable space. If the initial search region is unknown, enough diversity in the initial population should be guaranteed in order to explore a reasonably sized variable space before concentrating on the most promising regions.[35]

The term population refers to a set of candidate solutions. A single candidate solution is called an individual/chromosome x, which is a vector whose size depends on the number of variables to be optimized.[36]

The iterative process of evolution (Figure 3) is initialized with a population of randomly generated individuals. The population size N_{pop} is fixed in the design and does not depend on the size of the individuals. Once the initial population of N_{pop} chromosomes has been defined, each individual is evaluated in accordance with its associated fitness value, which is a measurement of its quality. A group of individuals has a higher probability to reproduce when their fitness, i.e., their ability to flourish in their environment, is high. The population is evolved to find possible solutions (i.e., the fittest individuals) by applying GA operators, which are typically crossover and mutation.

Selection operates as survival and choice of mates between parents. When the GA proceeds, recombination of genes gives way to new offsprings. The N_{pop} individuals in a given generation are listed in a ranking, where all new offspring are included. Only the top N_{pop} individuals are kept for mating and the rest are discarded to make room for the new offspring.

The first attempt to solve multiobjective optimization problems made use of a population that was divided in subpopulations of the same size. Each of them was responsible for only one objective in particular. This approach generally tended to give poor results, which were improved by the introduction of the concept of nondominated solutions or Pareto optimality.[37]

The main idea is to arrange the individuals, ranking them as a function of their degree of domination in the population. As to

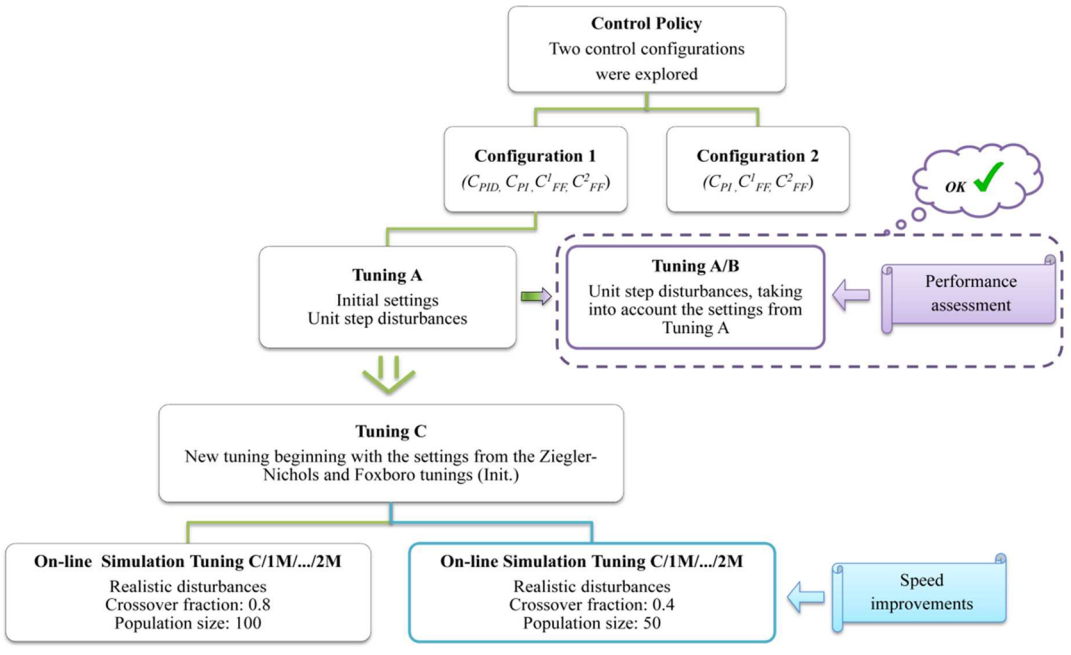


Figure 4. Design strategy.

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062−13073

the generation of the Pareto front, which is the line formed by the optimal solutions,[35,38] an algorithm was reported to maintain an archive of Pareto solutions. They proposed an evolutionary dynamic-weighted aggregation method, explaining how to change the weight dynamically during optimization. Moreover, stated various Pareto concepts showed that an EA statistically converges to a Pareto front.[39]

5. CONTROL STRATEGY

Figure 4 summarizes the design strategy followed in this work. First, two instances (with or without PID control) were explored with a view to selecting the most suitable arrangement in order to control the storage tank adequately. Second, for the adopted control policy (i.e., configuration 1, which includes a PID for feedback control), the controller settings were optimized with GAs.

As a fine-tuning improvement, the controllers were afterward tuned online to accommodate anomalies. First of all, starting from the settings obtained for configuration 1, unit step disturbances were introduced in order to assess the algorithmic performance when the GA output triggers changes in the controller settings. The responses without (keeping tuning A) and with (changing the settings from tuning A to tuning B) algorithmic intervention were evaluated. These tests showed that online adjustments are useful because the controllers behaved more satisfactorily when their settings were changed with tuning B.

Then, the starting point was polished to take into account realistic working conditions. In industrial practice, the controllers are conventionally set by means of traditional methods, like Ziegler–Nichols or Foxboro rules. Therefore, those values were adopted as a starting point (tuning C), and the GA speed was analyzed, trying to satisfy industrial requirements for online updating. To include realistic disturbances in the analysis, testing was carried out by considering uncertainties in the supply, taking the information from recent tunings 1M and 2M. More details of the results are given in Section 6.

5.1. Layout of the Control System. As shown in Figure 5, the system requiring control is the storage tank that is subjected

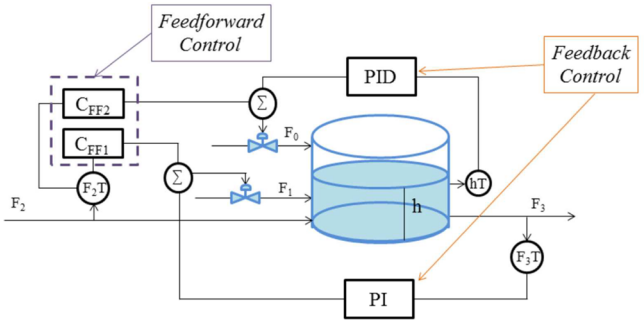


Figure 5. Layout of the system under control.

to three inputs. Two control variables associated with the tank are available: liquid level (h) and outflow (F_3). Unlike most of the typical multivariable control designs, it is proposed that this tank can be controlled unconventionally by means of a MIMO (Multiple-Input–Multiple-Output) system with two outputs and three inputs, instead of merely two. This is preferable when the production of natural gasoline declines. In the controller design, it was taken into account that the production profiles of a field tend to decline exponentially for several years before they can be considered depleted. This behavior arises from the mathematical expression (eq 1), which was developed by Mohr for the projection of oil and gas production.[32]

The purpose of the storage tank is to maintain sufficient inventory to feed the processing plants. The output flow should be as constant as possible, avoiding sudden changes. It should be taken into account that a tank input may be interrupted due to upstream conditions. Therefore, the control system should accommodate any normal input discrepancies.

Process control can be improved through multiple loops. The inclusion of a feedback signal actually adapts the forward loop to unmeasured changes in the process. For optimum performance, a feedback/feedforward system for liquid level and outflow, being both the controlled variables, ought to be supplied with updated information regarding well dynamics. Therefore, an adaptive control system is proposed. It has the ability to adjust controller settings in accordance with the changing character of the input flow rates. Controller settings are adapted online by using GAs.

The control-loop diagram of the entire system is illustrated in Figure 6, where $G(s)$ is the transfer function of the given plant, $C_{PI}(s)$is the transfer function of a PI controller, $C_{PID}(s)$ is the transfer function of a PID controller, $C^j_{FF}(s)$, j = 1,2 are the transfer functions of both feedforward controllers. Their Laplace transforms are reported in Table 2, where K_P^{PID}, τ_I^{PID}, and τ_D^{PID} are, respectively, the proportional, integral, and derivative settings of the PID controller, K_P^{PI} and τ_I^{PI} are, respectively, the proportional and integral settings of the PI controller, and K^j_{FF}, τ^j_1, and τ^j_2 are, respectively, the gain, manipulated, and load time constants of the jth feedforward controller. All these settings must be tuned to meet the prescribed performance criteria.

In Figure 6, the GA procedure is represented in dashed lines as a rectangular block. The errors are communicated to GA, which internally executes the optimization iteratively, making use of the input. When the process finishes, it returns the new controller settings.

5.2. MOEA Settings. The GA's driving force is the selection of individuals based on their fitness, which is a quality measure theoretically defined from operational requirements. The problem was modeled as a biobjective optimization problem (M = 2) by means of a Conventional Weighted Aggregation approach. The weighed sum of the objective functions (eq 3) was built by including the control objectives in the fitness of individuals as linear aggregated functions[33] (eq 5). The fitness function $F(x)$ was formulated (eq 5) aiming at a minimization of the system error. Since the errors of both controllers have different orders of magnitude, the natural choice was to specify the weights with a priori knowledge of the set points of the controlled variables.

For each objective, the fitness values were normalized by accounting for the deviations from the corresponding set points, with 0 being the best fitness. In eq 5, the errors are the difference between the measured values and the set points. For the PID controller, the error ε_{PID} is the departure of the measured level h_m from the required one h_{SP} (eq 6). Likewise, the error of the PI controller ε_{PI} is calculated (eq 7) by subtracting the measured flow from the tank outlet F_m, and the set-point outflow F_{SP} is required to be constantly sent to the processing plant.

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062–13073

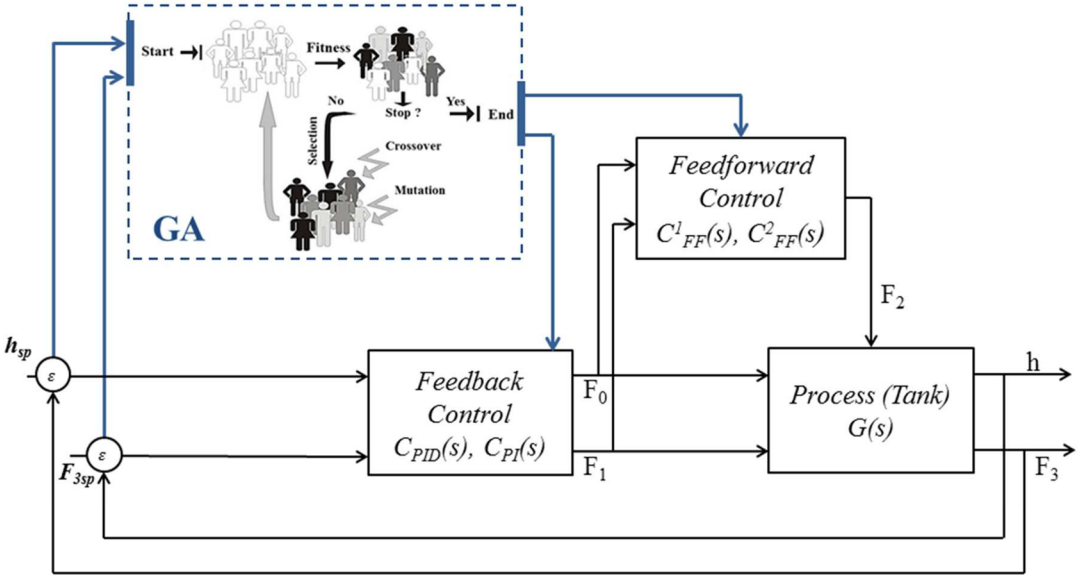


Figure 6. Structure of the adaptive system.

Table 2. Standard Transfer Functions

For the PID feedback controller: $C_{PID}(s) = K_P^{PID} + \tau_I^{PID}\frac{1}{s} + \tau_D^{PID}s$	C_{FF}^j = Transfer function of the jth feedforward controller, for j = 1, 2
	C_{PI} = Transfer function of the PI Controller
	C_{PID} = Transfer function of the PID Controller
For the PI feedback controller: $C_{PI}(s) = K_P^{PI} + \tau_I^{PI}\frac{1}{s}$	K_{FF}^j = Gain for the jth feedforward controller
	K_P^{PI} = Integral gain of the PI controller
	K_P^{PID} = Integral gain of the PID controller
For both feedforward controllers: $C_{FF}^j(s) = K_{FF}^j\frac{\tau_1^j s+1}{\tau_2^j s+1}$ j = 1, 2	τ_1^j, τ_2^j = Time constants for the jth feedforward controller
	τ_I^{PID}, τ_D^{PID}, τ_I^{PI} = Time constants for the feedback controllers

$$\boldsymbol{F}(\boldsymbol{x}) = \frac{|\varepsilon_{PID}|}{h_{SP}} + \frac{|\varepsilon_{PI}|}{F_{SP}} \tag{5}$$

where

$$f_1 = |\varepsilon_{PID}| = |h_m - h_{SP}| \tag{6}$$

$$f_2 = |\varepsilon_{PI}| = |F_m - F_{SP}| \tag{7}$$

$$\omega_1 = \frac{1}{h_{SP}} \tag{8}$$

$$\omega_2 = \frac{1}{F_{SP}} \tag{9}$$

For the GA termination criteria, both conditions stated in Section 4 were tested. For condition 1, NI_{max} = 150 was adopted. As to condition 2, the best fitness value $\hat{F}_g$ for the gth generation was monitored with a relative error tolerance TOL = 10 e^{-6}. In Figure 7, it is evident how the best fitness value decreases as the generations proceed.

As to the constraints in eq 2, D = 0, while C depends on the amount of controllers that are incorporated. For each element in x, there are always two inequalities associated with its

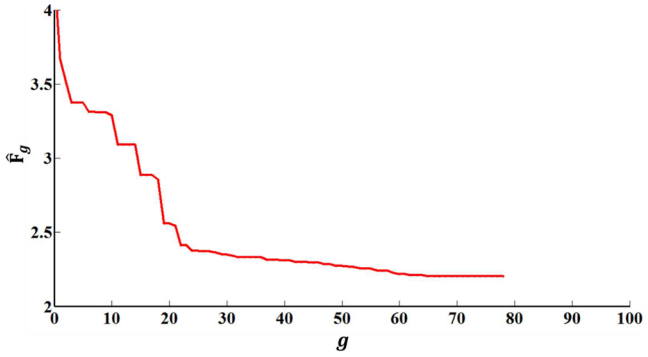


Figure 7. Evolution of the multiobjective optimization function.

bounds. Since N is the number of decision variables, the total number of constraints C can be calculated as $C = 2N$. The constraints $c_1(x),...,c_C(x) \leq 0$ are given by the upper and lower values of the individual settings of each PI and PID controller. In general, these bounds can be defined with the help of the controller settings derived from experimental observations[40] based on conventional well-known tuning methods, like Ziegler–Nichols[41] and Cohen–Coon[42] for PI and PID controllers and Foxboro tuning[43] for feedforward controllers. These settings are useful because they may not only serve as references to establish ranges to build an adequate search space but also as initialization values for the chromosomes in the initial population. The ith constraint c_i is associated (eq 10) to the upper bound x_i^{max} that is allowed for the ith element in x, while the $N + i$th constraint c_{N+i} is associated (eq 11) to the lower bound x_i^{min} that is allowed for the ith element in x.

$$c_i = x_i - x_i^{max} \leq 0 \quad i = 1, ...N \tag{10}$$

$$c_{N+i} = x_i^{min} - x_i \leq 0 \quad i = 1, ...N \tag{11}$$

The decision variables in $\boldsymbol{x} \in \mathbb{R}^N$ are the N settings of all the controllers, given in Table 2. It is more convenient to insert them as the elements of an individual in GA. In this work, special attention is given to the feedback loop. Two different configurations are tested in order to obtain the optimum

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062−13073

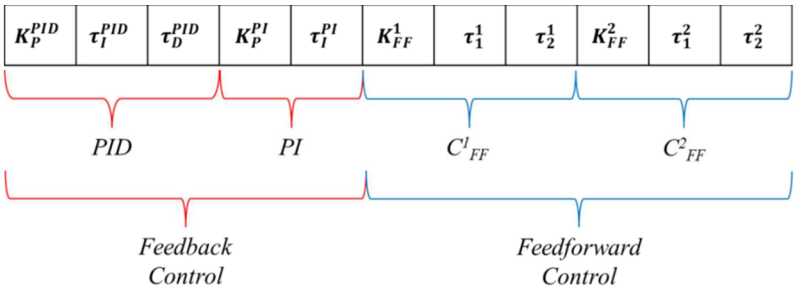


Figure 8. Representation of a GA individual/chromosome (configuration 1).

performance by identifying the more effective control system. Each of the configurations represents a different control policy with a distinct representation of individuals. In both cases, there is a feedforward controller, being the only difference in the feedback control strategy. For configuration 1, the chromosome representation is given by Figure 8, its size being $N = 11$. The first five elements are the settings of both a PI and a PID controller, and the next six elements are the settings of the feedforward controllers. In contrast, configuration 2 employs only one PI controller for the feedback loop. The structure of its chromosome is given in Figure 9, its size being $N = 8$.

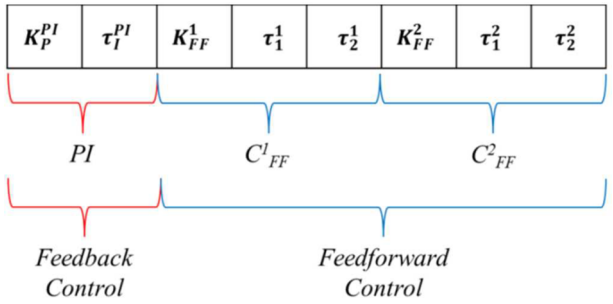


Figure 9. Representation of a GA individual/chromosome (configuration 2).

Selecting GA parameters like N_{pop} is very difficult due to the many possible variations in the algorithm and cost function. Since a GA relies on random number generators for creating the population, mating, and mutation, the task of comparing all the different options for a wide range of fitness functions is really hard. Moreover, the results may strongly depend on the fitness function analyzed. In particular, De Jong studied GA parameters intensively and concluded that a small population size improved initial performance, while large population size improved long-term performance.[44] Then, $50 \leq N_{pop} \leq 100$. This conclusion was drawn from tests on a suite of five fitness functions and testing problems with individuals of sizes 10 and 30. Later, Schaffer et al. added five more fitness functions to De Jong's test function suite.[45] They tested discrete sets of parameter values including $N_{pop} = 10$, 20, 30, 50, 100, and 200. These authors found the best online performance resulted for these settings: $20 \leq N_{pop} \leq 30$. For problems with moderately long chromosomes (lengths ranging from 30 to 60), Goldberg and Holland derived an expression for optimal population size (eq 12),[46] where N is the length of the chromosome, but Schaffer et al. suggested that this criterion is too conservative since it leads to extremely large populations.[45]

$$N_{pop} = 1.652^{0.2N} \tag{12}$$

6. RESULTS

6.1. Choice of Control Policy. Figure 5 shows only one possible control policy (configuration 1). Perhaps, the additional overhead introduced by a PID controller may not justify the resulting solutions' quality. Therefore, a single-objective optimization was tested by removing the PID device (configuration 2). As a result of a comparison between both systems, which is explained below, configuration 1 was finally adopted.

For the feedback/feedforward loops, the settings of the controllers were simultaneously optimized by means of a GA. In this work, two optimization stages were performed and tested. Figures 8 and 9 show the meaning assigned to each element of x for configurations 1 and 2, respectively. Since it is desired to find the best controller settings, gains and time constants naturally constitute the vector components.

In this application, 150 generations and scattered crossover were employed for all runs. Fixed combinations (N_{pop}, c_f) ranging from (100, 0.8) to (50, 0.4) were adopted for the tests. The elite solutions, which were the best children found so far, were destined to propagate unchanged, i.e., without being subjected to random mutations. This elitism is widespread in GAs because it is important to keep good answers during the evolution. The crossover fraction c_f is the fraction of each population that is made up of crossover children, excluding elite children.[47] The population was kept below 100 chromosomes since the individuals are always relatively short in this problem ($N = 11$ forconfiguration 1; $N = 8$ for configuration 2). In both cases, the goal was to solve an optimization problem where we searched for an optimal (minimum) solution in terms of the variables of the problem (the controller parameters to be optimized). The general program was built to generate the individuals relatively quickly. Therefore, the population size

Table 3. Mean Best Fitness Values and Optimal Settings for Controllers in Configurations 1 and 2

		C_{PID}		C_{PI}			$C^1{}_{FF}$			$C^2{}_{FF}$		
configuration	MBF	K_P^{PID}	τ_I^{PID}	τ_D^{PID}	K_P^{PI}	τ_I^{PI}	K_{FF}^1	τ_1^1	τ_2^1	K_{FF}^2	τ_1^2	τ_2^2
1	2.23	27.75	30.99	1.55	3.61	22.62	5.25	10.37	11.63	16.01	0.02	0.79
2	2.19	–	–	4.52	3.41	0.21	0.03	2.69	18.12	0.03	1.37	4.52

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062−13073

needed to be small so that the individuals could develop and evolve quickly.

It is important to average the results over multiple runs because the GA has many random components.[35] Besides, Cantú-Paz and Goldberg stated that multiple runs are more beneficial at very small population sizes.[48] Therefore, 30 executions were adopted for each search.

The executions were performed on an AMD 8120 Eight-Core Processor with 3.10 GHz and 8 GB of RAM. GA optimizations were run in a MATLAB environment, and the fitness values were obtained after having simulated by means of Simulink.[49,50] Mean Best Fitness Values (MBF) are reported in Table 3, together with the optimal individuals. For configuration 1, where the fitness formulation includes the PID objective, MBF is slightly higher than the value for configuration 2, where all errors are only due to the PI controller. Hence, more effective compensation is possible by leaving both controllers (PI and PID) at work.

The shape of the Pareto front was determined to evaluate the stability of the runs for the EAs.[38] The feedback control illustrated in Figure 4 was tested by applying a unit step in F_2. For this two-objective problem, Figure 10 shows the Pareto

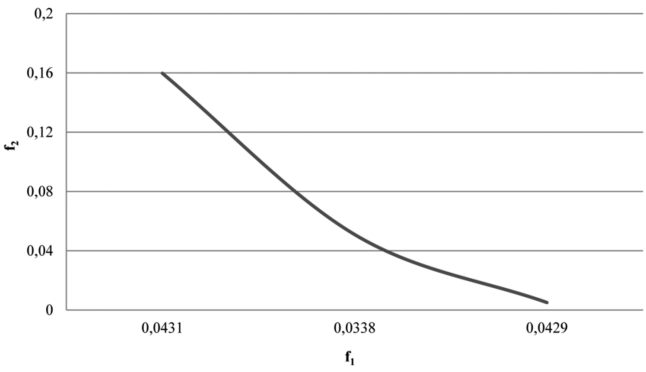


Figure 10. Convex Pareto front.

front obtained in the conventional way, where the objective function f_2 (eq 7) is usually plotted versus f_1 (eq 6). Since the corresponding Pareto Front is convex, each weight combination corresponds to a stable minimum on this front.

6.2. Online GA Parametric Adjustment. At this point, advanced process control was incorporated to face some operational challenges. The high degree of well-to-well variability makes it difficult to reliably assess the amounts that will be ready to store. Figure 6 illustrates an online, real-time solution that can compensate these transient imbalances. The proposed strategy consists in tailoring the controller settings by means of GAs. In this way, the tank feed can be maintained at optimal levels on the bases of actual well conditions.

There is large variability in the initial production rates of wells, and it is economically important to analyze early life well behavior and, hence, predict the future oil production.[32] Since there is additional uncertainty in the production decline rate for each field, it is proposed that the controllers are tuned online to accommodate anomalies. The procedure consists in starting with the controller settings for configuration 1 given by Table 3, and later, they are being updated on the basis of recent field-history records. To eliminate the effect of the measured field disturbances in F_2, the controllers are tuned again with the best results yielded by GA optimization, which is always running as an internal procedure. Then, remedial action can be taken on the basis of actual conditions, rather than predetermined settings for the controller settings.

Some experiments were carried out to assess the impact of this advanced control. Unit step disturbances were introduced as F_2 inputs at different time points in order to test how each tuning responds. For example, Figure 11 compares the

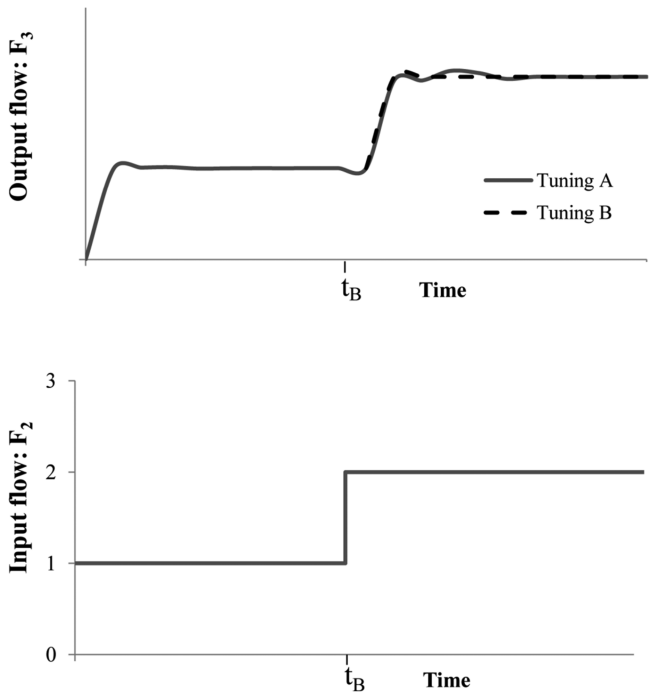


Figure 11. Comparison of system responses to unit step disturbances in input flow rate F_2.

responses of the most relevant outputs, i.e., tank level and output flow, without (tuning A) and with (tuning B) changes in the controller settings. Table 4 shows gains and time constants for both tunings. The settings were updated at t_B for tuning B. The policy of keeping tuning A after t_B is not advisable because it becomes less and less effective, exhibiting more severe overshoots. In contrast, tuning B contributes to responsiveness tolerating the changes. It can be concluded that the algorithmic adjustment can adapt the controllers satisfactorily at operational level.

6.3. About the Self-Tuning Adaptive Approach. With the aim of adapting the controllers (Figure 6) to the real environment, the system was tested as a whole by considering uncertain supplies at the operational level. To simulate these disturbances in a realistic way, they were predicted by applying an algorithm-based approach, which allows forecasting the dynamics of natural gasoline resources.[32] The model (eq 1) includes a number of settings that were fitted to historical production data, specifically comprising information from Argentina. Sur Piedra Clavada [(lat., long.) = (−46.65, −68.69)] is a particularly promising gas field in Argentina.[51] We have inferred its empirical behavior from the data collected from governmental reports.[26]

Figure 12 shows the predicted dynamic behavior for the field Sur Piedra Clavada (F_2), which was born in 2007. At about 2027, its exploitation may begin because this is the time when this field is expected to have reached maximum production.

Effective disturbance rejection was achieved under control by using the self-tuning adaptive approach. When the field's

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062−13073

Table 4. Mean Best Fitness Values and Optimal Settings for Both Tunings

		C_{PID}			C_{PI}		C^1_{FF}			C^2_{FF}		
tuning	MBF	K_P^{PID}	τ_I^{PID}	τ_D^{PID}	K_P^{PI}	K_P^{PID}	τ_I^{PID}	τ_D^{PID}	K_P^{PI}	K_P^{PID}	τ_I^{PID}	τ_D^{PID}
A	2.25	27.75	30.99	1.55	3.61	22.62	5.25	10.37	11.63	16.01	0.02	0.79
B	2.35	5.22	53.05	0.76	21.10	15.37	0.80	8.11	7.95	0.17	20.35	23.47

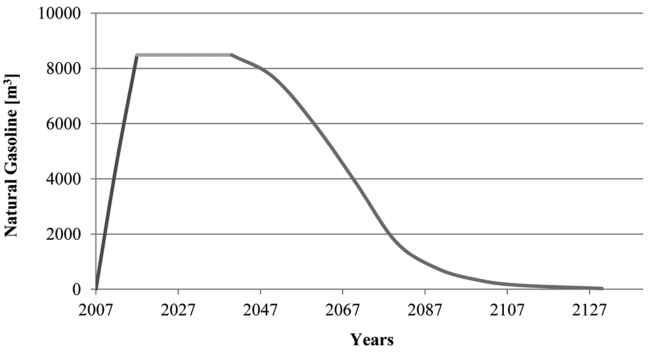


Figure 12. Field profile predicted for Sur Piedra Clavada.

dynamic behavior is given by Figure 12, the responses in a closed loop are satisfactory. The outputs react as shown in Figure 13, being the corresponding adjustments reported on

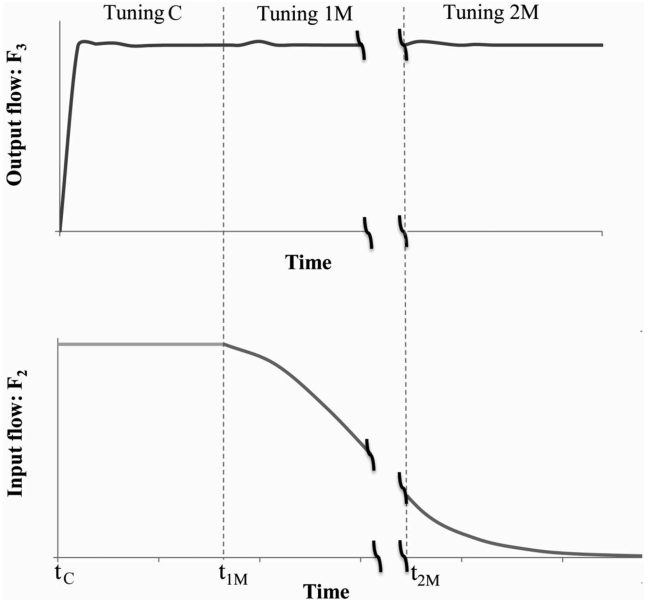


Figure 13. Simulation results using GA parametric adjustment.

Table 5. Tuning A, which corresponds to the peak production, has the initial settings for the controllers that start working at t_C. When the field starts to decline, its data are collected, and they are provided to the GA, whose runtimes to find a group of more accurate values are reported in Table 5.

When starting from the Ziegler–Nichols tuning for PID and PI settings and Foxboro tuning for the feedforward controllers, which were reported on the first row (init.) of Table 5, the computational time required to reach tuning C is 186 s. This computational time is valid only at the beginning of the adjustment procedure because it is taken by the MOEA to generate the complete population set. The MOEAs are useful to guide the search for online control, whenever their performance is fast enough. The internal procedures, which is the GA algorithm for the implementation proposed in this paper, have to be integrated in a relatively short period of time to ensure the online feasibility of control and optimization procedures.[52] From the viewpoint of online responsiveness in industrial practice, computational times ranging between 20 and 50 s are considered acceptable.[53] Besides, subsequent tunings will naturally start from previous settings, which implies that the GA population set is at that point rather close to the optimal values, thus requiring shorter times to calculate tuning readjustments. The time Δt = 122 s was taken to reach tuning 1M starting from the settings of tuning C, while the runtime that corresponds to the path from tuning 1M to tuning 2M was Δt = 102 s. Since the computational times are still higher than the values commonly accepted industrially, the GA algorithm should still be improved.

There is a compromise dilemma between the computational efforts and the prediction capacity. It should be considered that the runtimes reported in this paper were empirically taken by using the MATLAB environment,[47] which is a high-level interpreted programming language. Then, if it is desired to achieve lower computational times, the code should be improved in a certain way. In this sense, the code could either be reprogrammed by using compiled languages or else a parallel GA implementation could be designed because parallel programming enables fast convergence by reducing the number of iterations and execution times.[54] Nevertheless, before taking these extreme measures, it is advisible to make an effort to improve GA performance by handling GA settings. The regulation can be made taking into account experimental tests. The elementary genetic plans designed by De Jong[44] suggest that larger populations respond more slowly but yield better long-term performance. It is suggested that increasing the population size or the mutation rate may degrade online performance. Moreover, reducing the crossover fraction

Table 5. GA Performance Indicators

			C_{PID}			C_{PI}		C^1_{FF}			C^2_{FF}		
tuning	Δt [s]	MFB	K_P^{PID}	τ_I^{PID}	τ_D^{PID}	K_P^{PI}	τ_I^{PI}	K^1_{FF}	τ_1^1	τ_2^1	K^2_{FF}	τ_1^2	τ_2^2
init.	0	6.117	5.22	26.08	0.26	4.73	5.35	1.31	15.70	1.15	15.73	0.07	0.71
C	186	2.153	21.13	41.51	1.43	14.9	5.81	0.78	8.47	8.11	16.93	0.02	0.834
1M	122	2.169	21.13	41.50	1.43	15.39	5.81	0.76	9.91	0.31	17.21	0.02	0.933
2M	102	2.184	21.28	41.58	1.23	15.43	5.81	0.18	1.37	6.32	2.22	0.02	0.32

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062–13073

Table 6. Runtimes for GA Algorithm

			C_{PID}			C_{PI}		C^1_{FF}			C^2_{FF}		
tuning	Δt [s]	MFB	K_P^{PID}	τ_I^{PID}	τ_D^{PID}	K_P^{PI}	τ_I^{PI}	K_{FF}^1	τ_1^1	τ_2^1	K_{FF}^2	τ_1^2	τ_2^2
C	186	2.153	21.13	41.51	1.43	14.90	5.81	0.78	8.47	8.11	16.93	0.02	0.834
1M	42	2.173	21.30	41.60	1.23	15.43	5.81	0.18	1.37	6.32	2.22	0.02	0.32
2M	49	2.193	21.60	40.43	1.55	14.43	5.81	3.78	8.23	8.28	15.12	0.02	0.68

improves the initial performance because the sampling rate might be too high.

By making changes in the GA settings, which were originally set with a crossover fraction equal to 0.8 and a population size of 100 individuals, runtimes could be reduced to computational times closer to acceptable values, as reported on Table 6. According to the policy suggested by De Jong,[44] satisfactory runtimes were achieved when the crossover fraction was decreased up to 0.4 and the population size was reduced to 50 individuals. The speed improvements were due to the adequate changes in the settings that rule the genetic evolution. According to Table 6, the tests yield that updating the controller settings every 50 s is the minimum updating frequency that can be adopted with this implementation. Larger updating times should be avoided because violent control actions may also be introduced, and the GA will generally take a long time to find the best individual. On the other hand, if more speed is required, a more efficient implementation in a compiled language is advisable.

As to the control behavior, Figure 13, where the controllers were updated at times t_{1M} and t_{2M}, shows that the tracking performance proved to be effective. Moreover, the tank output flow rate was significantly improved by means of well-tuned controllers.

7. CONCLUSIONS

In this work, an MOEA for the self-tuning control of a storage tank fed from various natural gasoline sources is proposed. The optimization method is a Bi-Objective Evolutionary Algorithm based on a Genetic Algorithm, which was applied as a controller-tuning technique to ensure optimal control performance. Selecting the best combination of controller settings to exploit the dynamics of the problem proved to be advantageous because smoother and more effective control actions could be achieved.

The self-tuning procedure is a stable algorithm with a simple architecture. It is easy to implement and efficient to be used in real-world applications. The analysis of the behavior of the internal GA procedure showed that it is advisable to choose adequately the algorithmic settings to ensure fast adjustments. The experimental tests revealed that it is convenient to choose a small population size (about 50 individuals) and a low crossover fraction (about 0.4).

Testing was carried out by predicting the production profile of the main source from real data. The regulation method proved to be successful to avoid or reduce the impact of the dynamics of the fields. By suitable adjustments in settings and modifications to the basic genetic plan, a considerable improvement in performance was achieved. Less than 50 s were necessary to find an updated set of controller settings, even though the implementation can still be improved if higher computational speeds are required. As part of our future work, it would be interesting to reduce the computational efforts by implementing this MOEA with parallel programming techniques.

AUTHOR INFORMATION

Corresponding Author

*E-mail: dybrigno@criba.edu.ar. Tel: 054-0291-4861700, Int: 250.

ORCID

Paola Patricia Oteiza: 0000-0002-1138-5626

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the financial support from CONICET and MINCYT: Agencia Nacional de Promoción Científica y Tecnológica for the Grant Prest BID PICT 2012 0691 given by Fondo para la Investigación Científica y Tecnológica, Argentina.

NOMENCLATURE

$c_i(x)$ = ith inequality constraint, for $i = 1, C$
C = Number of inequality constraints
c_f = Crossover fraction
C_{FF}^j = Transfer function of the jth feedforward controller, for $j = 1, 2$
C_{PI} = Transfer function of the PI controller
C_{PID} = Transfer function of the PID controller
$d_i(x)$ = ith equality constraint, for $i = 1, D$
D = Number of equality constraints
Δt = Increment of runtimes (s)
D = Decision space
ϵ_g^F = The relative error of the best values of the multiobjective optimization function between two successive generations
ε_{PI} = Error of the PI controller
ε_{PID} = Error of the PID controller
$F(x)$ = Multiobjective optimization function. In particular, the fitness function for an MOEA with GA
$\hat{F}_g$ = Best value of the multiobjective optimization function for the gth generation
f_i = ith objective function of the optimization problem, for $i = 1, M$
F_i = Flow rate of the ith stream
F_m = Measured outflow from the tank
F_P = Maximum production plateau
F_{sp} = Set-point of the outflow from the tank
g = Index that corresponds to the generation
g_0 = Initial generation
$G(s)$ = Transfer function of the process
h = Liquid level of the tank
h_m = Measured level of the tank
h_{sp} = Set-point level of the tank
j = Index that identifies a feedforward controller
K_{FF}^j = Gain for the jth feedforward controller, a tuning setting
K_P^{PI} = Integral gain, a tuning setting for the PI controller
K_P^{PID} = Integral gain, a tuning setting for the PID controller

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062−13073

M = Number of objective functions
N = Number of decision variables
NI_{max} = Maximum number of iterations
N_{pop} = Population size
P = Population in the GA procedure
$P_F(t)$ = Fields production profile, as a function of time t
Q_r = Amount of recoverable resources remaining in the field at t_r
s = Laplace transform variable
τ_1^j, τ_2^j = Time constants for the jth feedforward controller, two tuning settings for each controller
τ_I^{PID}, τ_D^{PID}, τ_I^{PI} = Time constants for the feedback controllers
t_B = Updating time for tuning B (s)
t_C = Updating time for tuning C (s)
t_F = Total amount of time necessary to reach the maximum production plateau (year)
t_r = Moment when the field production begins to decline (year)
t_{1M} = Updating time for tuning 1M (s)
t_{2M} = Updating time for tuning 2M (s)
TOL = Lower bound on the change in the best value of the multiobjective optimization function during a step
ω_i = Non-negative weight for the ith objective, for $i = 1, M$
x = Vector of decision variables
x_i^{max} = Maximum upper value that is allowed for the ith element in x
x_i^{min} = Minimum lower value that is allowed for the ith element in x
Y_F = Time instant when the field commences the production (year)

REFERENCES

(1) Rodriguez Rubio, F.; Lopez Sánchez, M. J. *Control Adaptativo y Robusto*; Universidad de Sevilla: Spain, 1996.
(2) Shamsuzzoha, M. Closed-Loop PI/PID Controller Tuning for Stable and Integrating Processes. *Ind. Eng. Chem. Res.* **2013**, *52* (36), 12973−12992.
(3) Cho, W.; Lee, J.; Edgar, T. F. Simple Analytic Proportional-Integral-Derivative (PID) Controller Tuning Rules for Unstable Processes. *Ind. Eng. Chem. Res.* **2014**, *53* (13), 5048−5054.
(4) Li, T.; de Silva, C. W.; Mitra, S. Design of a Self-tuning Controller for Local Water Quality Adjustment. The 9th International Conference on Computer Science & Education (ICCSE 2014), 2014, pp 218−222.
(5) Zheng, Y. J.; Chen, S. Y.; Ling, H. F. Evolutionary Optimization for Disaster Relief Operations: A Survey. *Appl. Soft Comput.* **2015**, *27*, 553−566.
(6) Giagkiozis, I.; Fleming, P. J. Methods for multi-objective optimization: An analysis. *Inf. Sci.* **2015**, *293*, 338−350.
(7) Oh, W. S.; Sol, K.; Cho, K. M.; Yoo, K. S.; Kim, Y. T. Genetic based self-tuning speed controller for induction motor drives. *International Conference on Applied Superconductivity and Electromagnetic Devices (ASEMD)* **2011**, 88−91.
(8) Liu, C. H.; Hsu, Y. Y. Design of a Self-Tuning PI Controller for a STATCOM Using Particle Swarm Optimization. *IEEE T. Ind. .Electron* **2010**, *57*, 702−715.
(9) Kennedy, J. Particle Swarm Optimization. In *Encyclopedia of Machine Learning*; Springer, 2010.
(10) Viswanathan, P. K.; Toh, W. K.; Rangaiah, G. P. Closed-Loop Identification of TITO Processes Using Time-Domain Curve Fitting and Genetic Algorithms. *Ind. Eng. Chem. Res.* **2001**, *40*, 2818−2826.
(11) Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans. *Evol. Comput.* **2002**, *6*, 182−197.
(12) Behroozsarand, A.; Shafiei, S. Optimal control of amine plant using non-dominated sorting genetic algorithm-II. *J. Nat. Gas Sci. Eng.* **2010**, *2*, 284−292.
(13) Ayala, H. V. H.; dos Santos Coelho, L. Tuning of PID controller based on a multi-objective genetic algorithm applied to a robotic manipulator. *Expert. Syst. Appl.* **2012**, *39* (10), 8968−8974.
(14) Chang, W. D. A multi-crossover genetic approach to multivariable PID controllers tuning. *Expert. Syst. Appl.* **2007**, *33*, 620−626.
(15) Meng, X.; Song, B. Fast Genetic Algorithms Used for PID Parameter Optimization. *Proceedings of the IEEE International Conference on Automation and Logistics* **2007**, 2144−2148.
(16) Cha, Y.-J.; Raich, A.; Barroso, L.; Agrawal, A. Optimal placement of active control devices and sensors in frame structures using multi-objective genetic algorithms. *Struct. Control Health Monit.* **2013**, *20*, 16−44.
(17) Neath, M. J.; Swain, A. K.; Madawala, U. K.; Thrimawithana, D. J. An optimal PID controller for a bidirectional inductive power transfer system using multi-objective genetic algorithm. *IEEE T. Power. Electr.* **2014**, *29* (3), 1523−1531.
(18) Tochampa, W.; Sirisansaneeyakul, S.; Vanichsriratana, W.; Srinophakun, P.; Bakker, H. H. C.; Wannawilai, S.; Chisti, Y. Optimal control of feeding in fed-batch production of xylitol. *Ind. Eng. Chem. Res.* **2015**, *54*, 1992−2000.
(19) Manenti, F.; Manca, D. Transients modeling for enterprise-wide optimization: generalized framework and industrial case study. *Chem. Eng. Res. Des.* **2009**, *87*, 1028−1036.
(20) Kidnay, A. J.; Parrish, W. *Fundamentals of Natural Gas Processing*, 1st ed.; CRC Press Taylor & Francis Group: London, 2006.
(21) Countries by Natural Gas Proven Reserves. Wikipedia. https://en.wikipedia.org/wiki/List_of_countries_by_natural_gas_proven_reserves (accessed February 24, 2016).
(22) NGL Price Update: Natural Gasoline. Energy & Income Advisor. https://www.energyandincomeadvisor.com/ngl-price-update-natural-gasoline/ (accessed April 2, 2016).
(23) US Crude Oil Reserves Increase on 6-Year Trend. Oil & Gas Journal. http://www.ogj.com/articles/2015/11/us-crude-oil-reserves-increase-on-6-year-trend.html (accessed March 2, 2016).
(24) Reserves Grow Modestly as Crude Oil Production Climbs. Oil & Gas Journal. http://www.ogj.com/articles/print/volume-113/issue-12/special-report-worldwide-report/reserves-grow-modestly-as-crude-oil-production-climbs.html (accessed March 12, 2016).
(25) Instituto Petroquímico Argentino. Información Estadística de la Industria Petroquímica y Química de la Argentina, 34th ed.; IPA: Buenos Aires, Argentina, 2014.
(26) Secretaría de Energía. República Argentina. Producción de Petróleo y Gas (Tablas Dinámicas). http://www.energia.gov.ar/contenidos/verpagina.php?idpagina=3299 (accessed March 23, 2016).
(27) Cañete, B.; Oteiza, P. P.; Gigola, C. E.; Brignole, N. B. Gasolina Natural: un sustituto atractivo para la producción de etileno en Argentina. Revista Petroquímica, Petróleo Gas y Química, http://revistapetroquimica.com/gasolina-natural-un-sustituto-atractivo-para-la-produccion-de-etileno-en-argentina/. 2012, *283*, 206−201 (accessed November 2, 2016).
(28) García Sánchez, A. *Programación del transporte de hidrocarburos por oleoductos mediante la combinación de técnicas metaheurísticas y simulación. Escuela Técnica Superior de Ingenieros Industriales*. Thesis, Universidad Politécnica de Madrid, Spain, 2007.
(29) Sullivan, F. M.; Montgomery, J. B. The Salient Distribution of Unconventional Oil and Gas Well Productivity; MITEI-WP-2015-05; MIT Energy Initiative Working Paper; MIT Energy Initiative: Cambridge, MA, 2015.
(30) Zhang, J.; Sun, Z.; Zhang, Y.; Sun, Y.; Nafi, T. Risk-opportunity analyses and production peak forecasting on world conventional oil and gas perspectives. *Pet. Sci.* **2010**, *7* (1), 136−146.
(31) Maggio, G.; Cacciola, G. When will oil, natural gas, and coal peak? *Fuel* **2012**, *98*, 111−123.

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062−13073

(32) Mohr, S. Projection of World Fossil Fuel Production with Supply and Demand Interactions. Ph.D. Thesis in Engineering (Chemical), The University of Newcastle, Australia, 2010.
(33) Coello Coello, C. A.; Lamont, G. B.; Van Veldhuizen, D. A. *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd ed.; Springer: New York, 2007.
(34) Darwin, C. *On the Origin of Species by Means of Natural Selection*; John Murray: London, 1859.
(35) Haupt, R. L.; Haupt, S. E. *Practical Genetic Algorithms*; John Wiley & Sons: Hoboken, NJ, 2004.
(36) Luke, S. *Essentials of Metaheuristics*, 2nd ed.; Lulu.com: USA, 2013.
(37) Perez Serrada, A. *Proyecto fin de carrera, Una introducción a la Computación Evolutiva*; Universidad de Valladolid, Valladolid, España, 1996.
(38) Jin, Y.; Olhofer, M.; Sendhoff, B. Dynamic Weighted Aggregation for Evolutionary Multi-Objective Optimization: Why Does It Work and How? *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001)*, 2001; pp 1042−1049.
(39) Van Veldhuizen, D. A.; Lamont, G. B. Evolutionary Computation and Convergence to a Pareto Front. *Late Breaking Papers at The Genetic Programming 1998 Conference*, 1998; pp 221−228.
(40) Shamsuzzoha, M. Closed-loop PI/PID controller tuning for stable and integrating process with time delay. *Ind. Eng. Chem. Res.* **2013**, *52* (36), 12973−12992.
(41) Ziegler, J. G.; Nichols, N. B. Optimum settings for automatic controllers. *Trans ASME* **1942**, *64*, 759−768.
(42) Noris, M. F. B. M. M.S. Thesis, Comparison between Ziegler−Nichols and Cohen−Coon Method for Controller Tunings. Faculty of Chemical & Natural Resources Engineering, Bachelor of Chemical Engineering University College of Engineering & Technology, Malaysia, 2006.
(43) LeBlanc, S. E.; Coughanowr, D. R. *Process Systems Analysis and Control*, 3rd ed.; McGraw-Hill: New York, 2009.
(44) De Jong, K. A. An Analysis of the Behavior of a Class of Genetic Adaptive Systems. Doctoral Dissertation, University of Michigan: Ann Arbor, MI, 1975.
(45) Schaffer, J. D.; Caruana, R. A.; Eshelman, L. J.; Das, R. A Study of Control Parameters Affecting On-Line Performance of Genetic Algorithms for Function Optimization, *Proceedings of the 3rd International Conference on Genetic Algorithm*, 1989; pp 51−60.
(46) Goldberg, D. E.; Holland, J. H. Genetic algorithms and machine learning. *Mach Learn.* **1988**, *3* (2), 95−99.
(47) Vary Mutation and Crossover. MathWorks. https://www.mathworks.com/help/gads/vary-mutation-and-crossover.html?requestedDomain=www.mathworks.com&requestedDomain=www.mathworks.com (accessed November 1, 2016).
(48) Cantú-Paz, E.; Goldberg, D. E. Are multiple runs of Genetic Algorithms better than one?, GECCO 2003, PT I. *Proceedings lecture notes in Computer Science* **2003**, *2723*, 801−812.
(49) Chipperfield, A.; Fleming, P.; Pohlheim, H.; Fonseca, C. *Genetic Algorithm TOOLBOX for Use with MATLAB*; University of Sheffield: South Yorkshire, U.K., 1994.
(50) Introduction: Simulink Modeling. Control Tutorials for MATLAB & SIMULINK. http://ctms.engin.umich.edu/CTMS/index.php?example=Introduction&section=SimulinkModeling (accessed February 8, 2016).
(51) Fundación YPF. http://energiasdemipais.educ.ar/mapa/ (accessed May 8, 2016).
(52) Lima, N.; Linan, L. Z.; Maciel, M. R. W.; Embiruçu, M.; Grácio, F.; Filho, R. M. Modeling and predictive control using fuzzy logic: application for a polymerization system. *AIChE J.* **2010**, *56* (4), 965−978.
(53) Lima, N. M.; Liñan, L. Z.; Manenti, F.; Embiruçu, M.; Ferreira, E. C.; Maciel, M. R. W. Multivariable Nonlinear Advanced Control of Copolymerization Processes. *20th European Symposium on Computer Aided Process Engineering-ESCAPE20*, 2010; pp 1−6.
(54) Alba, E. *Parallel Metaheuristics: A New Class of Algorithms*; Wiley, 2005.

DOI: 10.1021/acs.iecr.6b01934
Ind. Eng. Chem. Res. 2016, 55, 13062−13073