Hybrid genetic pattern search augmented Lagrangian algorithm: application to WWTP optimization

Isabel A.C.P. Espírito Santo[1,2], Lino Costa[1,2], Roman Denysiuk[2] and Edite M.G.P. Fernandes[1,2]

[1] Department of Production and Systems
University of Minho, Campus de Gualtar, 4710-057 Braga, Portugal
{iapinho,lac,emgpf}@dps.uminho.pt
[2] Algoritmi R&D Center, Portugal
denysiukr@gmail.com

Abstract. An augmented Lagrangian algorithm is presented to solve a global optimization problem that arises when modeling the activated sludge system in a Wastewater Treatment Plant, attempting to minimize both investment and operation costs. It is a heuristic-based algorithm that uses a genetic algorithm to explore the search space for a global optimum and a pattern search method for the local search refinement. The obtained results have physical meaning and show the effectiveness of the proposed method.

1 Introduction

The high costs associated with the design of Wastewater Treatment Plants (WWTP) have motivated research in the area of process modeling and optimization of the treatment water processes. This paper is part of an ongoing research project in which we are engaged to optimize the design and operation of WWTP in terms of minimum total cost (investment and operation costs). The model herein presented focuses only on the secondary treatment, the one that mostly affects the WWTP design. This search for a minimum is a challenge, due to the complexity of the model. In order to find the global minimum of the WWTP problem, we propose a Hybrid Genetic Pattern Search Augmented Lagrangian (HGPSAL) algorithm that hybridizes a genetic algorithm with a derivative-free pattern search method to refine the best solution found by the genetic search. Equality and inequality constraints of the problem are handled by an augmented Lagrangian framework.

Penalty function methods provide powerful theoretical and computational tools for the study of constrained optimization problems. Many engineering design problems can be formulated as constrained optimization problems. The presence of constraints has been affecting the performance of most optimization algorithms for unconstrained problems. Constraint-handling techniques, in particular when embedded with stochastic algorithms for solving difficult nonconvex real optimization problems, provide today a hot topic of research. The most

popular technique to handle constraints is indeed the penalty function method. The simple principle and the easy implementation motivated the popularity of the method. The approach augments the function that is being optimized by adding in weighted terms that incorporate the constraints. A penalty parameter is used to balance the search for the global optimum and the satisfaction of the constraints. The constrained problem is transformed into unconstrained ones. Besides interior penalty functions that are not well suited for an evolutionary framework, due to the mutation operator, the exterior penalty functions have been widely used in the literature [2, 19, 21]. The two most common penalty functions involve absolute and quadratic penalty terms. While the absolute penalty function is exact for a nonzero value of the penalty parameter, the corresponding penalty approach produces a narrow valley floor in the search space, with a 'V' shape. On the other hand, the quadratic penalty approach tends to produce a wider valley floor, with a 'U' shape, to search for the solution. However, it has been shown that the quadratic penalty approach only converges to the optimal solution of the constrained problem if the penalty parameter is driven to zero. Numerically, the problem becomes more ill-conditioning as the penalty parameter approaches zero. An augmented Lagrangian is a more sophisticated penalty function for which a nonzero penalty parameter value is sufficient to yield convergence to the solution of the constrained problem [3]. Furthermore, an augmented Lagrangian framework provides a wider valley floor search space. The augmented Lagrangian functions depend on the Lagrange multiplier vector, which ought to be updated as the technique approaches the optimal solution. There are different formulae for updating the multiplier vector. It can be shown that the overall rate of convergence depends on the rate of convergence of the variables, and of the multiplier vector. When Newton's method is used to solve the unconstrained problems, second order update formulae for the multiplier vector should be used so that the quadratic nature of the entire process is retained. However, since genetic algorithms may be considered zero order methods, any updating formula for the multiplier vector is suitable.

In this paper, we address the implementation of an augmented Lagrangian method combined with a genetic algorithm for solving a particularly difficult engineering design problem. Further, a derivative-free local search procedure, like the one proposed in [17], is incorporated into the iterative process to refine the search in the neighborhood of the solution. This work has been motivated by other papers published in the area of augmented Lagrangian functions [4–6, 18]. Birgin *et al.* [5] combine an augmented Lagrangian approach with a deterministic global optimization method (the αBB method) and its convex α-underestimation. Conn *et al.* [6] and Lewis and Torczon [18] presented modified augmented Lagrangian methods for nonconvex optimization with equality constraints and proved global convergence results without assuming that the sequence of multiplier vectors is bounded. Methods based on augmented Lagrangians are common in local search methods [6, 9, 16, 18] and in deterministic type methods for global optimization, for example in [4, 5, 20], but rare when combined with heuristics that rely on a population of points to converge to the solution [1, 22, 23].

2 Hybrid genetic pattern search augmented Lagrangian algorithm

In order to solve the problem, an algorithm based on a hybridization of an augmented Lagrangian with a genetic algorithm is presented. Further, the hybridization of global and local optimizers provides a more effective tradeoff between exploitation and exploration of the search space. It is well-known that overall successful and efficient general solvers do not exist. Stochastic population based algorithms like genetic algorithms [13] are good at identifying promising areas of the search space (exploration), but less good at fine-tuning approximations to the minimum (exploitation). Conversely, local search algorithms like pattern search are good at improving approximations to the minimum. This algorithm aims to find a global solution of an optimization problem in the form:

$$
\begin{aligned}
& \underset{x \in \Omega}{\text{minimize}} \quad f(x) \\
& \text{subject to } b_i(x) = 0, \quad i = 1, \ldots, m \\
& \hphantom{\text{subject to }} g_j(x) \leq 0, \quad j = 1, \ldots, p
\end{aligned}
\tag{1}
$$

where x is an n dimensional vector and $\Omega = \{x \in \mathbb{R}^n : l \leq x \leq u\} \subset \mathbb{R}^n$, $f(x)$ is the objective function, $b(x) = 0$ are the equality constraints and $g(x) \leq 0$ are the inequality constraints. The herein implemented augmented Lagrangian function is

$$
\Phi(x; \lambda, \delta, \mu) = f(x) + \lambda^T b(x) + \frac{1}{2\mu} \|b(x)\|_2^2 + \frac{\mu}{2} \sum_{i=1}^{p} \left(\max\left\{ 0, \delta_i + \frac{g_i(x)}{\mu} \right\}^2 - \delta_i^2 \right)
$$

where μ is a positive penalty parameter, $\lambda = (\lambda_1, \ldots, \lambda_m)^T$ and $\delta = (\delta_1, \ldots, \delta_p)^T$ are the Lagrange multiplier vectors associated with the equality and inequality constraints respectively. Function Φ aims to penalize solutions that violate the equality and inequality constraints only. The simple bounds $l \leq x \leq u$ are not integrated into the penalty terms. Hence, the bound constrained subproblem has the form:

$$
\underset{x \in \Omega}{\text{minimize}} \ \Phi^j(x) \equiv \Phi(x; \lambda^j, \delta^j, \mu^j).
\tag{2}
$$

The solution of (2) for each set of fixed λ^j, δ^j and μ^j, gives an approximation to the solution of (1). We refer to [3, 11] for details. Denote this approximation by x^{j+1}. Here the index j is the counter of the outer iterative process. The Lagrange multipliers λ^j and δ^j are estimated in this iterative process according to the well-known first-order updating formulae,

$$
\lambda_i^{j+1} = \max\left\{ \lambda_{\min}, \min\left\{ \lambda_i^j + \frac{b_i(x^{j+1})}{\mu^j}, \lambda_{\max} \right\} \right\}, \ i = 1, \ldots, m,
\tag{3}
$$

$$
\delta_i^{j+1} = \max\left\{ 0, \min\left\{ \delta_i^j + \frac{g_i(x^{j+1})}{\mu^j}, \delta_{\max} \right\} \right\}, \ i = 1, \ldots, p,
\tag{4}
$$

where a safeguard scheme is used to maintain the multiplier vectors bounded throughout the process, $\lambda_i^j \in [\lambda_{\min}, \lambda_{\max}]$, $i = 1, \ldots, m$, $\delta_i^j \in [0, \delta_{\max}]$, $i =$

$1, \ldots, p$. It is expected that the sequence of penalty parameters $\{\mu^j\}$ decreases although very small values may cause numerical difficulties when solving subproblem (2). Our proposal defines a lower bound $\mu_{\min}$ as follows:

$$\mu^{j+1} = \max \left\{ \min \left\{ \mu^j \nu, \left(\mu^j \right)^{\theta_\mu} \right\}, \mu_{\min} \right\}, \ 0 < \nu \le 1, \ 0 < \theta_\mu \le 1. \tag{5}$$

The traditional augmented Lagrangian methods are locally convergent if the subproblems (2) are solved according to a certain tolerance, herein denoted by ε^j [18]. Further, when $j \to \infty$, the sequence $\{\varepsilon^j\}$ should approach zero. To define these tolerances, the idea in [18] is extended to include the multipliers associated with the inequality constraints:

$$\varepsilon^j = \tau \left(1 + \|\lambda^j\| + \|\delta^j\| + (\mu^j)^{-1} \right)^{-1}, \ 0 < \tau < 1. \tag{6}$$

Note that a decreasing sequence of μ values will cause a decreasing sequence of ε values if the multiplier vectors have a reasonable behavior. The general augmented Lagrangian algorithm for solving problem (1) is the following.

Algorithm 1 *HGPSAL Algorithm*

1. *Given $0 < \tau < 1$, $\lambda_{\min} < \lambda_{\max}$, $\delta_{\max} > 0$, $\mu_{\min} \ll 1$, $\lambda_i^0 \in [\lambda_{\min}, \lambda_{\max}]$, $i = 1, \ldots, m$, $\delta_i^0 \in [0, \delta_{\max}]$, $i = 1, \ldots, p$, $\mu^0 > 0$, $0 < \theta_\mu \le 1$, $0 < \nu \le 1$, $x^0 \in \Omega$, $l_{\max}$, set $j = 0$;*
2. *Compute ε^0 using (6);*
3. *While the stopping criterion is not met do*
   3.1. Find an ε^j approximate minimizer x^{j+1} to the subproblem (2) using Algorithm 2;
   3.2. Update λ^{j+1} using (3), δ^{j+1} using (4) and μ^{j+1} using (5);
   3.3. Compute ε^{j+1} using (6);
   3.4. Set $j = j + 1$;

The algorithm stops when the current objective function value improves 80% over the objective function value at x^0, and the constraints violation improves 99% over the violation at x^0. A maximum number of iterations $l_{\max}$ is also defined in case the stopping criterion is not met. The herein proposed technique for solving (2) uses a population based algorithm, known as genetic algorithm, followed by a local search procedure. The general form for the bound constrained algorithm implemented in this paper is shown below.

Algorithm 2 *Hybrid Genetic Pattern Search Bound Constrained Algorithm*

1. *Given $x^j \in \Omega$;*
2. *Find $y^j \leftarrow GA(x^j)$, using the genetic algorithm presented in Subsection 2.1;*
3. *Find $x^{j+1} \leftarrow HJ(y^j)$, using the Hooke and Jeeves version of the pattern search algorithm described in Subsection 2.2.*

We remark that since the genetic algorithm is a population based method, y^j is the point with best fitness found by the algorithm. Details concerning each step of the algorithm are presented below.

2.1 Genetic algorithm for global search

A Genetic Algorithm (GA) is a population based algorithm that uses techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover [13]. Thus, unlike conventional algorithms, GAs start from a population of points P of size s. In spite of the traditional binary representation used by GAs, in our implementation, a real representation is used since we are leading with a continuous problem. Therefore, each point of the population $z^{(l)}$, for $l = 1, \ldots s$, is an n dimensional vector. A fitness function is defined as evaluation function in order to compare the points of the population and to apply a stochastic selection that guarantees that better points are more likely to be selected. The fitness function corresponds to the objective function of the subproblem (2), i.e., $\Phi^j(x)$. New points in the search space are generated by the application of genetic operators (crossover and mutation) to the selected points from population. Elitism was implemented by maintaining, during the search, a given number e, of the best points in the population.

Crossover combines two points in order to generate new ones. A Simulated Binary Crossover (SBX) [7] that simulates the working principle of single-point crossover operator for binary strings was implemented. Two points, $z^{(1)}$ and $z^{(2)}$, are randomly selected from the pool and, with probability p_c, two new points, $w^{(1)}$ and $w^{(2)}$ are generated according to

$$
\begin{aligned}
w_i^{(1)} &= 0.5\left((1+\beta_i)z_i^{(1)} + (1-\beta_i)z_i^{(2)}\right) \\
w_i^{(2)} &= 0.5\left((1-\beta_i)z_i^{(1)} + (1+\beta_i)z_i^{(2)}\right),
\end{aligned}
\qquad
\beta_i =
\begin{cases}
(2r_i)^{\frac{1}{\eta_c+1}} & \text{if } r_i \le 0.5 \\
\left(\frac{1}{2(1-r_i)}\right)^{\frac{1}{\eta_c+1}} & \text{if } r_i > 0.5
\end{cases}
\tag{7}
$$

for $i = 1, \ldots, n$, where $r_i \sim U(0,1)$ and $\eta_c > 0$ is an external parameter of the distribution. This procedure is repeated until the number of generated points equals the number of points in the pool.

A Polynomial Mutation is applied, with a probability p_m, to the points produced by the crossover operator. Mutation introduces diversity in the population since crossover, exclusively, could not assure the exploration of new regions of the search space. This operator guarantees that the probability of creating a new point $t^{(l)}$ closer to the previous one $w^{(l)}$ $(l = 1, \ldots, s)$ is more than the probability of creating one away from it. It can be expressed by:

$$
t_i^{(l)} = w_i^{(l)} + (u_i - l_i)\iota_i, \quad
\iota_i =
\begin{cases}
(2r_i)^{\frac{1}{\eta_m+1}} - 1 & \text{if } r_i < 0.5 \\
1 - (2(1-r_i))^{\frac{1}{\eta_m+1}} & \text{if } r_i \ge 0.5
\end{cases}
\tag{8}
$$

for $i = 1, \ldots, n$, where $r_i \sim U(0,1)$ and $\eta_m > 0$ is an external parameter of the distribution. The GA proceeds as the following algorithm.

Algorithm 3 *Genetic Algorithm*

1. *Given x^j, $e > 1$, $s > 1$, $0 < p_c < 1$, $\eta_c > 0$, $0 < p_m < 1$, $\eta_m > 0$, $k_\Delta \ge 1$, $k_{\max} > 1$, set $k = 0$;*
2. *Set $z^{(1),k} = x^j$ and randomly generate $z^{(l),k} \in \Omega$, for $l = 2, \ldots, s$*

3. While the stopping criterion is not met do

    3.1. Compute $\Phi^j(z^{(l),k})$, for $l = 1, \ldots, s$

    3.2. Select by tournaments $s - e$ points from P

    3.3. Apply SBX crossover to the $s - e$ points, with probability p_c using (7)

    3.4. Apply mutation to the $s - e$ points with probability p_m using (8)

    3.5. Replace the worst $s - e$ points of P

    3.6. Set $k = k + 1$;

4. Set $y^j = z^k_{best}$;

This procedure stops when $|\Phi^j(z^k_{best}) - \Phi^j(z^{k-k_\Delta}_{best})| \leq \varepsilon^j$ where $\Phi^j(z^k_{best})$ is the fitness value of the best point in population, at iteration k, and k_Δ is a parameter that defines a periodicity for testing the criterion. However, if the previous criterion is not satisfied in $k_{\max}$ iterations, the procedure is terminated and the best point in the population is returned.

2.2 Pattern search algorithm for local search

A pattern search method is a derivative-free method that performs, at each iteration k, a series of exploratory moves along the coordinate axes around a current approximation, z^k, in order to find a new approximation $z^{k+1} = z^k + \Delta^k s^k$, with a lower fitness value. We use k for the iteration counter of this inner iterative process. For $k = 0$, the initial approximation to begin the search is $z^0 = y^j$ (see Algorithm 2). The scalar Δ^k represents the step length and the vector s^k determines the direction of the step. The exploratory moves to produce $\Delta^k s^k$ and the updating of Δ^k and s^k define a particular pattern search method and their choices are crucial to the success of the algorithm. The iteration is considered successful if $\Phi^j(z^{k+1}) < \Phi^j(z^k)$; otherwise it is unsuccessful. When an iteration is successful, the step length is not modified, while in an unsuccessful iteration, Δ^k is reduced.

The step $\Delta^k s^k$ is computed by the Hooke and Jeeves (HJ) search method [15]. When the previous iteration was successful and z^k was accepted as the new approximation, a pattern move is defined by $z^k - z^{k-1}$. A new trial approximation is then defined as $z^k + (z^k - z^{k-1})$ and an exploratory move is then carried out around this trial point. If this search is successful, the new approximation is accepted as z^{k+1}. We refer to [15, 17] for details. A new approximation to the problem (1), $x^{j+1} \leftarrow z^{k+1}$ is provided if the condition $\Delta^k \leq \varepsilon^j$ holds. However, if this stopping condition does not hold for $k_{\max}$ iterations, then the procedure is stopped with the last available approximation. The main steps of this local search are displayed in Algorithm 4. We note that any infeasible approximation generated by both Algorithms 3 and 4 is projected onto the set Ω [17].

Algorithm 4 *Pattern Search Algorithm*

1. Given y^j, ε^j, Δ^0, $k_{\max} > 1$, set $k = 0$;

2. Set $z^k = y^j$;

3. While the stopping criterion is not met do

    3.1. Compute $\Delta^k s^k$ using HJ exploratory moves such that $z^k + \Delta^k s^k \in \Omega$;

3.2. If $\Phi^j(z^k) - \Phi^j(z^k + \Delta^k s^k) > 0$ then $z^{k+1} = z^k + \Delta^k s^k$ else $z^{k+1} = z^k$;

3.3. Update Δ^k and s^k;

3.4. Set $k = k + 1$;

4. Set $x^{j+1} = z^k$.

3   The case study: WWTP optimal design

The problem of economically and at the same time optimally designing the activated sludge system of a WWTP amounts to determining the values of the variables such that the investment and operation costs associated with the design of the WWTP are minimal. All the other constraints that have to be considered during the design process should also be taken into consideration.

3.1   Mathematical model of the system

The system under study consists of an aeration tank, where the biological reactions take place, and a secondary settler for the sedimentation of the sludge and clarification of the effluent. To describe the aeration tank we chose the activated sludge model n.1, described by Henze et al. [14], which considers both the elimination of the carbonaceous matter and the removal of the nitrogen compounds. A set of constraints arises from the mass balances around the aeration tank using the Peterson matrix of the ASM1 model [14]. The generic equation for a mass balance around a certain system considering a completely stirred tank reactor (CSTR) is

$$\frac{Q}{V_{\mathrm{a}}}\left(\xi_{\mathrm{in}} - \xi\right) + r_\xi = \frac{d\xi}{dt} \ , \tag{9}$$

where Q is the flow that enters the tank, V_{a} is the aeration tank volume, ξ and ξ_{in} are the concentrations of the component around which the mass balances are being made inside the reactor and on entry, respectively. It is convenient to refer that in a CSTR the concentration of a compound is the same at any point inside the reactor and at the effluent of that reactor. The reaction term for the compound in question, r_ξ, is obtained by the sum of the product of the stoichiometric coefficients, $\nu_{\xi j}$, with the expression of the process reaction rate, ρ_j, of the ASM1 Peterson matrix [14] $r_\xi = \sum_j \nu_{\xi j} \rho_j$. In steady state, the accumulation term in (9) given by $\frac{d\xi}{dt}$ is zero, because the concentration is constant in time. A WWTP in labor for a sufficiently long period of time without significant variations can be considered at steady state. As our purpose is to make cost predictions in a long term basis it is reasonable to do so. The ASM1 model involves 8 processes incorporating 13 different components, such as the substrate, the bacteria, dissolved oxygen, among others. For the sake of clearness, we include here the mass balance equation related to one of the components - the soluble substrate (S_{S}):

$$\frac{-\mu_{\mathrm{H}}}{Y_{\mathrm{H}}} \frac{S_{\mathrm{S}}}{K_{\mathrm{S}} + S_{\mathrm{S}}} \left(\frac{S_{\mathrm{O}}}{K_{\mathrm{OH}} + S_{\mathrm{O}}} + \frac{\eta_{\mathrm{g}} K_{\mathrm{OH}}}{K_{\mathrm{OH}} + S_{\mathrm{O}}} \frac{S_{\mathrm{NO}}}{K_{\mathrm{NO}} + S_{\mathrm{NO}}} \right) X_{\mathrm{BH}} + \frac{k_{\mathrm{h}} X_{\mathrm{BH}}}{K_{\mathrm{X}} X_{\mathrm{BH}} + X_{\mathrm{S}}}$$

$$\left(\frac{S_\mathrm{O}}{K_\mathrm{OH} + S_\mathrm{O}} + \frac{\eta_\mathrm{h}\, K_\mathrm{OH}}{K_\mathrm{OH} + S_\mathrm{O}} \frac{S_\mathrm{NO}}{K_\mathrm{NO} + S_\mathrm{NO}} \right) X_\mathrm{S} + \frac{Q}{V_\mathrm{a}} \left(S_{S_\mathrm{in}} - S_\mathrm{S} \right) = 0 \, .$$

We denote all the soluble components by $S_?$ and the particulates by $X_?$. All the other symbols are stoichiometric or kinetic parameters for the wastewater considered. (See [14] for details on how to obtain all the other equations.)

Another set of constraints is concerned with the secondary settler. When the wastewater leaves the aeration tank, the treated water should be separated from the biological sludge, otherwise, the chemical oxygen demand would be higher than it is at the entry of the system. The ATV design procedure [8] contemplates the peak wet weather flow (PWWF) events, during which there is a reduction in the sludge concentration. To turn around this problem, a certain depth (h_3) is allocated to support the fluctuation of solids during these events. This way a reduction in the sedimentation area (A_s) is allowed. A compaction zone (h_4) where the sludge is thickened in order to achieve the convenient concentration to return to the biological reactor, also has to be contemplated and depends only on the characteristics of the sludge. A clear water zone (h_1) and a separation zone (h_2) should also be considered and are set empirically. The depth of the settling tank, h, is the sum of these four zones. The sedimentation area is still related to the peak flow, Q_p. The described relations are the following:

$$h_3 = \Delta X \, V_\mathrm{a} \frac{DVSI}{480 \, A_\mathrm{s}}, \quad h_4 = X_\mathrm{p} \frac{DVSI}{1000}, \quad \frac{Q_\mathrm{p}}{A_\mathrm{s}} \leq 2400 \, (X_\mathrm{p} DVSI)^{-1.34}$$

where $DVSI$ is the diluted volumetric sludge index and ΔX is the variation of the sludge concentration inside the aeration tank in a PWWF event.

Besides the ATV procedure, the double exponential model [24] is also used to describe the sedimentation process [10]. This model assumes a one dimensional settler, in which the tank is divided into 10 layers of equal thickness. It assumes that no biological reactions take place, meaning that the dissolved matter concentration is maintained across all the layers. Only vertical flux is considered and the solids are uniformly distributed across the entire cross-sectional area of the feed layer $(j = 7$, in our case). This model is based on a traditional solids flux analysis but the flux in a particular layer is limited by what can be handled by the adjacent layer. The settling function is given by

$$v_{s,j} = \max \left(0, \min \left(v_0', v_0 \left(e^{-r_h(TSS_j - f_\mathrm{ns}\, TSS_7)} - e^{-r_p(TSS_j - f_\mathrm{ns}\, TSS_7)} \right) \right) \right)$$

where $v_{s,j}$ is the settling velocity in layer j (m/day), TSS_j is the total suspended solids concentration in each of the ten considered layers of the settler and v_0, v_0', r_h, r_p and f_ns are settling parameters [24]. The solids flux due to the bulk movement of liquid may be up or down, v_up and v_dn respectively, depending on its position relative to the feed layer, thus $A_\mathrm{s} v_\mathrm{up} = Q_\mathrm{ef}$ and $A_\mathrm{s} v_\mathrm{dn} = Q_\mathrm{r} + Q_\mathrm{w}$. As to the subscripts, 'r' is concerned with the recycled sludge, 'w' corresponds to the wasted sludge and 'ef' is the treated effluent. The sedimentation flux, J_s, for the layers under the feed layer $(j = 7, \ldots, 10)$ is given by $J_{s,j} = v_{s,j} TSS_j$,

and above the feed layer $(j = 1, \ldots, 6)$ the clarification flux, J_{clar}, is given by

$$J_{clar,j} = \begin{cases} v_{s,j} TSS_j, & \text{if } TSS_{j+1} \leq TSS_t \\ \min\left(v_{s,j} TSS_j, \ v_{s,j+1} TSS_{j+1}\right) & \text{otherwise,} \end{cases}$$

where TSS_t is the threshold concentration of the sludge. The resulting solids balances around each layer, considering steady state, are the following:

$$v_{up}\left(TSS_2 - TSS_1\right) - J_{clar,1} = 0,$$
$$v_{up}\left(TSS_{j+1} - TSS_j\right) + J_{clar,j-1} - J_{clar,j} = 0, \ j = 2, \ldots, 6,$$
$$\left(Q\, TSS_7\right)/A_s + J_{clar,6} - \left(v_{up} + v_{dn}\right) TSS_7 - \min\left(J_{s,7}, \ J_{s,8}\right) = 0,$$
$$v_{dn}\left(TSS_{j-1} - TSS_j\right) + \min\left(J_{s,j}, \ J_{s,j-1}\right) - \min\left(J_{s,j}, \ J_{s,j+1}\right) = 0, \ j = 8, 9,$$
$$v_{dn}\left(TSS_9 - TSS_{10}\right) + \min\left(J_{s,9}, \ J_{s,10}\right) = 0.$$

The other important group of constraints are a set of linear equalities and define composite variables. In a real system, some state variables are, most of the time, not available for evaluation. Thus, readily measured composite variables are used instead. For example, the chemical oxygen demand (COD) is composed by soluble and particulate components, that are related by the equation

$$COD = S_I + S_S + X_I + X_S + X_{BH} + X_{BA} + X_P \ .$$

Similar equations can be defined for the volatile suspended solids (VSS), total suspended solids (TSS), biochemical oxygen demand (BOD), total nitrogen of Kjeldahl (TKN) and total nitrogen (N).

The system behavior, in terms of concentration and flows, may be predicted by balances. In order to achieve a consistent system, these balances must be done around the entire system and not only around each unitary process. They were done to the suspended matter, dissolved matter and flows. The equations for particulate compounds (organic and inorganic) have the following form

$$\left(1 + r\right) Q_{inf} X_{?_{ent}} = Q_{inf} X_{?_{inf}} + \left(1 + r\right) Q_{inf} X_? - \frac{V_a X_?}{SRT X_{?_r}}\left(X_{?_r} - X_{?_{ef}}\right) - Q_{inf} X_{?_{ef}}$$

and the equations for the solubles are $\left(1 + r\right) Q_{inf} S_{?_{in}} = Q_{inf} S_{?_{inf}} + r Q_{inf} S_{?_r}$, where r is the recycle rate, SRT is the sludge retention time and $Q_?$ represents the volumetric flows. The subscripts in the previous equations mean: 'inf' represents the influent wastewater, 'ent' is related to the entry of the aeration tank, 'r' is the recycled sludge and 'ef' represents the treated effluent.

It is also necessary to add some system variables definitions, in order to define the system correctly. In this group we include the sludge retention time, the recycle rate, hydraulic retention time (HRT), recycle rate in a PWWF event (r_p), recycle flow rate in a PWWF event (Q_{r_p}) and maximum overflow rate $\left(\frac{Q_p}{A_s}\right)$:

$$SRT = \frac{V_a X}{Q_w X_r}, \ HRT = \frac{V_a}{Q}, \ r = \frac{Q_r}{Q_{inf}}, \ r_p = \frac{0.7\, TSS}{TSS_{max_p} - 0.7\, TSS},$$
$$Q_{r_p} = r_p Q_p, Q_p \leq 2 A_s \ .$$

A fixed value of 0.7 for the relation between volatile (VSS) and total suspended solids (TSS) was considered. All the variables are considered nonnegative, although more restricted bounds are imposed to some of them due to operational consistencies. For example, the dissolved oxygen has to be always greater or equal to 2 mg/L. These conditions define a set of simple bounds on the variables. Finally, the quality of the effluent has to be imposed. The quality constraints are usually derived from law restrictions. The most used are related with limits in the COD, N and TSS at the effluent. In mathematical terms, these constraints are defined by portuguese laws as $COD_{\mathrm{ef}} \leq 125$, $N_{\mathrm{ef}} \leq 15$ and $TSS_{\mathrm{ef}} \leq 35$.

3.2 The objective function

The objective function of the problem represents the total cost and includes both investment and operation costs. The operation cost is usually on annual basis, so it has to be updated to a present value using the adequate economic factors of conversion. Each term in the objective function is based on the basic model $C = aZ^b$ [25], where a and b are the parameters to be estimated, C is the cost and Z is the characteristic of the unitary process that most influences the cost. For example, for the investment cost of the aeration tank, the volume (V_{a}) and air flow (G_{S}) are considered. The parameters a and b are estimated by the least squares technique, using real data collected from a WWTP building company. The operation cost of the aeration tank considers the air flow, and the investment and operation costs of the secondary settler depend on the sedimentation area, A_{s}, and the depth, h. Summing up all these terms, we get the following objective cost function:

$$f(V_{\mathrm{a}}, G_{\mathrm{S}}, A_{\mathrm{s}}, h) = 174.2V_{\mathrm{a}}^{1.07} + 12487G_{\mathrm{S}}^{0.62} + 114.8G_{\mathrm{S}} + 955.5A_{\mathrm{s}}^{0.97} + 41.3\left(A_{\mathrm{s}}h\right)^{1.07} \quad .$$

4 Numerical results and conclusion

The mathematical model has 71 parameters, 113 variables, 103 equality constraints and one inequality constraint. All the variables are bounded below and above. The stoichiometric, kinetic and operational parameters are the default values presented in the GPS-X simulator [12], and they are usually found in real activated sludge based plants. The HGPSAL algorithm was coded in Mat-

Table 1. Results of 20 runs for the WWTP problem

	f_{best}	f_{avg}	f_{worst}	standard deviation	function evaluations
HGPSAL	0.6677	1.0352	1.2221	0.1294	1035201
GA	2.6728	8.6111	15.1273	4.4016	1001000
HJ	1.0653	-	-	-	151906

Lab programming language and the numerical results were obtained with a Intel Core2 Duo CPU 1.8GHz with 2GB of memory. All parameter of HGPSAL algorithm were kept constant for all the executions. The parameters of the augmented Lagrangian used in the experiments are: $\lambda_{\min} = -10^{12}$, $\lambda_{\max} = 10^{12}$, $\delta_{\max} = 10^{12}$, $\mu_{\min} = 10^{-12}$, $\tau = 0.5$, $\theta_{\mu} = 1$, $\nu = 0.5$, $\mu^0 = 1$, $\lambda_i^0 = 1, \forall i$,

Table 2. Obtained optimal values for the most important variables

V_a	G_S	A_s	h	COD	TSS	N
1168	100	94	4.7	40.6	35	15

$\delta_i^0 = 1, \forall i$ and $j_{\max} = 200$. The Genetic algorithm parameters are: $s = 200$, $e = 20$, $p_c = 0.9$, $\eta_c = 20$, $p_m = 1/n$, $\eta_m = 20$ and $k_\Delta = 20$. The initial step length Δ^0 in Algorithm 4 is set to the initial value, x^0, provided by the GPS-X simulator [12] with the real influent data. The chosen values for the upper bounds of the Lagrange multipliers have no significant effect on the performance of the algorithm as long as they are sufficiently large. The same is true for the lower bounds set to the penalty parameter and multiplier vector λ. Several tests were done in order to choose appropriate values for the other parameters. The maximum number of allowed iterations $k_{\max}$ for Algorithms 3 and 4 is set to 100 iterations. We also solve the problem without hybridization, i.e., using the GA and the HJ, separately. Due to the stochastic nature of the HGPSAL and GA algorithms, we ran the problem 20 times. For the HJ solution, we only present f_{best}, since the algorithm is deterministic. Table 1 shows the results in terms of the best (f_{best}), the average (f_{avg}) and the worst (f_{worst}) objective function values obtained after the 20 runs, in millions of euros. The table also displays the standard deviation of the function values and the average number of function evaluations. The best solution is obtained by the HGPSAL algorithm, therefore, this seems to highlight the advantages of using hybridization to explore more efficiently the search space. Table 2 lists the optimal values for the most important decision variables of the best HGPSAL solution, namely, the aeration tank volume, the air flow, the area and depth of the secondary settler, as well as COD, TSS and N at the effluent. As to the effluent quality, the law limits were accomplished, in particular, the COD at the effluent is below the law limit, showing the robustness of the solution. The obtained physical dimensions of the units (aeration tank and secondary settler) are as expected. Although the obtained WWTP design is economically attractive, a multi-objective mixed-integer programming approach will be proposed in future designs to get better effluents quality and smaller investment and operation costs.

References

1. Barbosa HJC (1999). A coevolutionary genetic algorithm for constrained optimization. In Proceedings of the *1999 Congress on Evolutionary Computation*, DOI:10.1109/CEC.1999.785466. V 3. pp 1605–1611
2. Barbosa HJC and Lemonge ACC (2008) An adaptive penalty method for genetic algorithms in constrained optimization problems. In: H. Iba H (ed.) *Frontiers in Evolutionary Robotics* ISBN: 978-3-902613-19-6) I-Tech Education Publ., Austria
3. Bertsekas DP (1999) *Nonlinear Programming*. 2nd edn. Athena Scientific, Belmont
4. Birgin EG, Castillo R and Martínez JM (2004) Numerical comparison of Augmented Lagrangian algorithms for nonconvex problems. *Computational Optimization and Applications* 31:31–56
5. Birgin EG, Floudas CA and Martínez JM (2009) Global minimization using an Augmented Lagrangian method with variable lower-level constraints. *Mathematical Programming, Ser. A* DOI:10.1007/s10107-009-0264-y

6. Conn AR, Gould NIM, Sartenaer A and Toint PhL (1996) Convergence properties of an augmented Lagrangian algorithm for optimization with a combination of general equality and linear constraints. *SIAM Journal on Optimization* 6:674–703

7. Deb K and Agrawal RB (1995) Simulated binary crossover for continuous search space. *Complex Systems* 9:115–149

8. Ekama GA, Barnard JL, Günthert FW, Krebs P, McCorquodale JA, Parker DS and Wahlberg EJ (1978) *Secondary Settling Tanks: Theory, Modeling, Design and Operation*. Technical Report 6, IAWQ - International Association on Water Quality

9. Di Pillo G, Liuzzi G, Lucidi S and Palagi L (2003) An exact augmented Lagrangian function for nonlinear programming with two-sided constraints. *Computational Optimization and Applications* 25:57–83

10. Espírito Santo IACP, Fernandes EMGP, Araújo MM and Ferreira EC (2006) On the Secondary Settler Models Robustness by Simulation. *WSEAS Transactions on Information Science and Applications* 3:2323–2330

11. Fiacco AV and McCormick GP (1968) *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*. John Wiley & Sons

12. GPS-X Technical Reference (2002). Hydromantis, Inc.

13. Goldberg D (1989) *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley

14. Henze M, Grady Jr CPL, Marais GVR and Matsuo T (1986) *Activated Sludge Model no. 1*, Technical Report 1, IAWPRC Task Group on Mathematical Modelling for Design and Operation of Biological Wastewater Treatment

15. Hooke R and Jeeves TA (1961) Direct search solution of numerical and statistical problems. *Journal on Associated Computation* 8:212–229

16. Huang XX and Yang XQ (2003) A unified augmented Lagrangian approach to duality and exact penalization. *Mathematics of Operations Research* 28:533–552

17. Lewis RM and Torczon V (1999) Pattern search algorithms for bound constrained minimization. *SIAM Journal on Optimization* 9:1082–1099

18. Lewis RM and Torczon V (2002) A globally convergent augmented Lagrangian pattern search algorithm for optimization with general constraints and simple bounds. *SIAM Journal on Optimization* 12:1075–1089

19. Liu J-L and Lin J-H (2007) Evolutionary computation of unconstrained and constrained problems using a novel momentum-type particle swarm optimization. *Engineering Optimization* 39:287–305

20. Luo H, Sun X and Wu H (2008) Convergence properties of augmented Lagrangian methods for constrained global optimization. *Optimization Methods and Software* 23:763–778

21. Petalas YG, Parsopoulos KE and Vrahatis MN (2007) Memetic particle swarm optimization. *Annals of Operations Research* 156:99–127

22. Sedlaczek K and Eberhard P (2007) Augmented Lagrangian particle swarm optimization in mechanism design. *Journal of System Design and Dynamics* 1:410–421

23. Smith S (2002) Using evolutionary algorithms incorporating the augmented Lagrangian penalty function to solve discrete and continuous constrained non-linear optimal control problems. In: Collet P *et al.* (eds.). *Lecture Notes in Computer Science* V 2310. Springer-Verlag pp 295–308

24. Takács I, Patry GG and Nolasco D (1991) A Dynamic Model of the Clarification-Thickening Process, *Water Research* 25:1263–1271

25. Tyteca D, Smeers Y and Nyns EJ (1977) Mathematical Modeling and Economic Optimization of Wastewater Treatment Plants. *CRC Critical Reviews in Environmental Control* 8:1–89