22

Swarm Intelligence for Optimization in the Urban Water Industry

Joaquín Izquierdo, Idel Montalvo, Rafael Pérez-García & Carlos D. Alonso
Centro Multidisciplinar de Modelación de Fluidos
Universidad Politécnica de Valencia
Camino de Vera, s/n, 46022, Valencia
Spain

1. Introduction

Many engineering design problems can be cast as optimization problems. The urban water industry is no exception. Design is necessary to implement new configurations, improve existing systems, continue satisfying consumer needs, and to expand to meet new conditions. In this context, we consider the design of new Water Distribution Systems (WDS), and Wastewater Systems (WWS); as well as the rehabilitation and enlargement of existing systems. Taking into account the uncertainty of much of the data on problems in existing configurations, it is frequently necessary to solve difficult inverse problems where optimization techniques are of paramount importance. The calibration, identification, and detection of leaks in a WDS, which are truly important problems in the water industry, can be addressed as optimization problems. Due to increasing urban development, optimisation represents a permanent source of challenge for the management of many resources, especially water. The challenge of planning, designing, and managing urban water systems is increasingly difficult because various systems are needed, and new systems are constantly being developed; while existing systems need enlargement and rehabilitation. Furthermore, there is a great deal of concern regarding the search for mechanisms of sustainable water supply at a reasonable cost.

In this chapter, Particle Swarm Optimization (PSO), a well-established evolutionary optimization technique, is applied to problems in the urban water industry. Originally designed to deal with continuous variables, the PSO variant considered in this chapter overcomes three typical weaknesses in this optimization technique. Firstly, it is adapted to consider mixed discrete-continuous optimization as the problems we consider involve the use of both continuous and discrete variables. Secondly, one of the main drawbacks associated with PSO, namely, the difficulty in maintaining good levels of population diversity, and balanced local and global searches, is overcome. This formulation finds optimum, or near-optimum, solutions much more efficiently; and with considerably less computational effort, because it introduces a richer population diversity. Requiring fewer generations is a major advantage in real water distribution or wastewater systems, where cost and time constraints prohibit repeated runs of an algorithm and hydraulic evaluations. Finally, the cumbersome aspect, common to all metaheuristics, of choosing the right parameter values is tackled through self-adaptive and dynamic parameter control. The variant herein proposed is applied to: (i) designing a WDS; (ii) designing wastewater networks; and (iii) calibrating and identifying leaks in a WDS. These are only three among the huge pool of optimization problems that can be addressed by the proposed PSO variant. This technique has also provided excellent convergence characteristics, and good final solutions when applied to different real-world problems.

2. Antecedents

Optimization in water systems is considerably more difficult than simulation of these systems. Typically, optimization is a constrained nonlinear search problem involving both continuous and discrete variables. Thus, the problem is a mixed continuous and discrete constrained nonlinear optimization problem that is often highly dimensional. There may be many local optima in the search space; and there is no single optimization model, or search algorithm, for solving the problem without compromising solution accuracy, computational efficiency, and problem completeness.

Classical methods of optimization involve the use of gradients, or higher-order derivatives of the fitness function. However, they are not well suited for many real-world problems because they cannot process inaccurate, noisy, discrete, and complex data. Therefore, robust methods of optimization are often required to generate suitable results.

During the last decade, many researchers in the water field have shifted direction, leaving aside traditional optimization techniques based on linear and nonlinear programming, and embarking on the implementation of evolutionary algorithms: Genetic Algorithms (Savic & Walters, 1997; Wu & Simpson, 2001; Matías, 2003; Wu & Walski, 2005); Ant Colony Optimization (Maier et al., 2003; Zecchin et al., 2005); Simulated Annealing (Cunha & Sousa, 1999); Shuffled Complex Evolution (Liong & Atiquzzaman, 2004); and Harmony Search (Geem, 2006), among others.

Particle Swarm Optimization (PSO) is one of the evolutionary algorithms that has shown great potential and good perspectives for the solution of various optimization problems (Dong et al., 2005; Herrera et al., 2009; Janson et al., 2008; Jin et al., 2007; Izquierdo et al., 2008a; Izquierdo et al., 2008b; Liao et al., 2007; Montalvo et al., 2008b; Pan et al., 2007). The PSO algorithm was developed by (Kennedy & Eberhart, 1995) and is a multi-agent optimization system inspired by the social behaviour of a group of migrating birds trying to reach an unknown destination. The aim of this chapter is to show that this algorithm, with several modifications, can be used to find solutions for several optimization problems in urban water systems. PSO is similar to other evolutionary techniques in that it does not guarantee the global optimum; and may prematurely converge to local optima, especially in complex multi-modal search problems. Nevertheless, PSO can be easily implemented, and is computationally inexpensive, since memory and CPU speed requirements are low.

The structure of the chapter is as follows. We first briefly describe three kinds of important problems in the water industry, namely, the design of WDS, the design of WWS, and the calibration and identification of leaks in a WDS. Secondly, we present a mixed continuous-discrete variant of PSO endowed with an enriched diversity feature, which greatly improves the performance of conventional PSO. This variant can also avoid the cumbersome task of parameter selection because it uses a self-adapting technique managed by the algorithm itself. Finally, we show the results of specific applications to selected case-studies, some of which are well-known benchmark problems in the literature. In the case of the design of WDSs, a real-world water distribution network is also considered, which shows the ability of the presented technique to solve real-world problems.

3. The addressed problems

In this section, different problems in the water industry are considered, namely, the design of WDSs and WWSs, and the calibration and identification of leaks in a WDS. These problems have already been addressed using various optimization techniques by other authors (Mariles & Nava, 2007; Botrous et al., 2000; Martínez, 2007; Zecchin et al., 2006), among others.

3.1 The design of a WDS

The optimal design of a WDS consists in determining the values of all the involved variables in such a way that the investment and maintenance costs of the system are minimal, subject to a number of constraints (Izquierdo et al., 2004). A general strategy for solving the optimal design problem of a WDS involves the balancing of several factors: finding the lowest costs for layout and sizing using new components; reusing or substituting existing components; creating a working system configuration that fulfils all water demands; adhering to the design constraints; and guaranteeing a certain degree of reliability for the system (Goulter & Coals, 1986; Goulter & Bouchart, 1990).

The benchmark cases we initially consider have been used traditionally in the literature, and are standard examples used to demonstrate the application of a wide range of tests and analyses. The fitness function that has been traditionally used takes only pipeline costs into account. Nevertheless, a generalization to broader classes of fitness functions is straightforward, as shown below. Hence, in order to facilitate comparisons with the results obtained by other authors for these benchmark cases, we start by using the following fitness function to estimate the costs:

$$F(D) = \sum_{i=1}^{P} C(D_i) \cdot L_i \,. \tag{1}$$

P is the number of pipes in the network, $D = (D_i)$ is the vector of pipe diameters (which is P-dimensional and its components belong to a discrete set of commercially available diameters), $C(D_i)$ is the unit cost per unit length of diameter D_i , and L_i is the length of the i-th pipe. It should be noted that $C(\cdot)$ is a nonlinear function of diameter.

To restrict ourselves to the same rules used in the literature to deal with the benchmark problems, only three kinds of constraints are considered here: continuity equations and energy equations (strongly nonlinear) enforced in the hydraulic model; and lack of satisfaction of minimum pressures at demand nodes. Accordingly, the total cost of the network is considered as the sum of the network cost (1), and a penalty cost. This total cost is defined as

$$F = \sum_{i=1}^{P} C(D_i)\cdot L_i + \sum_{j=1}^{K} p_j \cdot v_j \; . \tag{2}$$

K is the number of constraints, $v_j = (P_{\min} - P_j)\cdot \mathrm{H}(P_{\min} - P_j)$, where $\mathrm{H}(\cdot)$ is the Heaviside step function, is the j-th constraint violation; and p_j represents the penalty parameter corresponding to constraint j with a large value to ensure that infeasible solutions have a cost greater than any feasible solution.

In addition, the distribution of flowrates through the network, and the piezometric head values, must satisfy the classical equations of continuity and energy enforced in the hydraulic model. The complete set of equations may be written, by using block matrix notation (Izquierdo et al., 2004), as

$$\begin{pmatrix} A_{11}(q) & A_{12} \\ A_{12}^{t} & 0 \end{pmatrix}\begin{pmatrix} q \\ H \end{pmatrix} = \begin{pmatrix} -A_{10}H_f \\ Q \end{pmatrix}, \tag{3}$$

where A_{12} is the so-called connectivity matrix that describes the way demand nodes are connected through the lines. Its size is $P \times N_p$ with N_p being the number of demand nodes and P the number of lines; q is the vector of the flowrates through the lines; H the vector of unknown heads at demand nodes; A_{10} is an $P \times N_f$ matrix, with N_f being the number of fixed head nodes with known head H_f ; and Q is the N_p-dimensional vector of demands. Finally, $A_{11}(q)$ is an $P \times P$ diagonal matrix. System (3) is a nonlinear problem, whose solution is the state vector $x = (q,H)^t$ of the system. Continuity and energy equations are enforced by the use of EPANET2 (Rossman, 2000), which is the benchmark hydraulic analysis tool used worldwide.

The first case considered is the New York Tunnel water supply network (see Fig. 1, left), which has been examined several times in the literature (Savic & Walters, 1997; Matías, 2003; Maier et al., 2003). A complete detailed description can be seen in (Dandy et al., 1996). The system has a fixed head reservoir, 21 tunnels, and 19 nodes. The objective of the New York Tunnel (NYT) problem is to determine the most economically effective design for adding to the existing system of tunnels forming the primary water distribution system for the city of New York. Because of age and increased demands, the existing gravity flow tunnels were found to be inadequate to meet the pressure requirements for the projected consumption level. The construction of additional gravity flow tunnels parallel to the existing tunnels was considered. All 21 tunnels are considered for duplication. There are 15 available discrete diameters, and one extra possible decision, which is the 'do nothing' option. The second considered case is the Hanoi pipe network (Fig. 1, right), also studied extensively by various researchers (Savic & Walters, 1997; Matías, 2003; Zecchin et al., 2005; Cunha & Sousa, 1999; Zecchin, 2003). The complete setting can be found in (Wu & Simpson, 2001). This network consists of a single fixed head source at an elevation of 100m, 34 pipes, and 31 demand nodes organized in three loops and two ramified branches. The objective is to specify the diameters (from a set of six commercially available diameters) for the 34 pipes, so that the total cost of the network is minimal, and the pressure at each node of consumption is at least 30m.

The problems faced in the optimal design of WDSs are considerable. Furthermore, this simple variant for the design of a WDS is NP-hard. The NYT system, with 21 pipes, and 15 potential commercial pipe diameters, has 16^{21} possible pipe diameter combinations (including the null option) that constitute the search space of the problem. The Hanoi problem, with 34 pipes and 6 potential pipe diameters, has 6^{34} possible pipe diameter combinations. These modest networks would require a considerable amount of time for an exhaustive search algorithm to navigate the entire search space of almost $2{\cdot}10^{25}$ and $2.87{\cdot}10^{26}$ potential solutions, respectively.

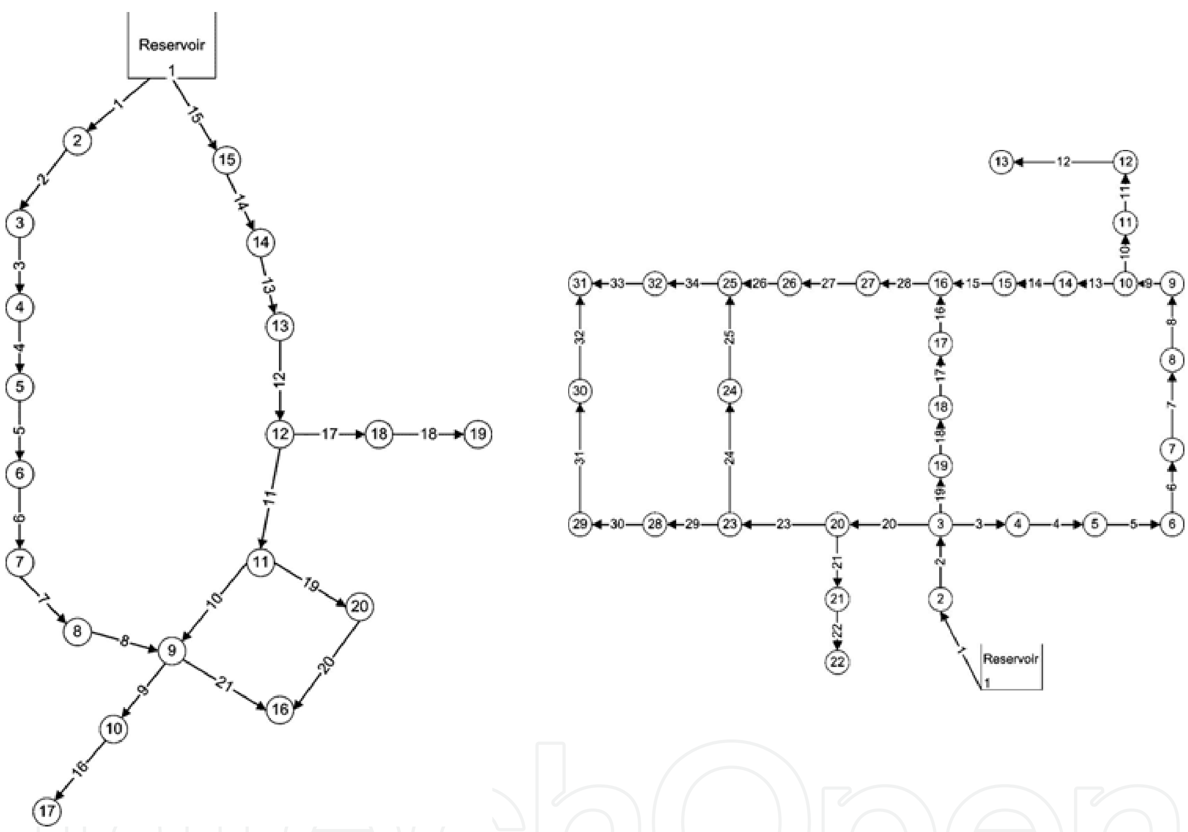


Fig. 1. NYT (left) and Hanoi (right) water distribution networks

In the case of real-world problems, design must consider other aspects, in particular, reliability. The term reliability refers to the ability of the network to provide consumers with adequate and high quality supply, under normal and abnormal conditions. Both hydraulic and mechanical reliability are considered. The former refers to uncertainty resulting mainly from nodal demand and pipe roughness. The latter usually refers to failures of system components, such as pipe breakage. However, there is no universal agreement about the best measure of reliability, redundancy, or resilience; nor what are acceptable levels for these concepts (see, for example, (Savic, 2005)). In this chapter, we considered a proposal recently raised in (Martínez, 2007), since it enforces a certain level of reliability on a system by considering costs incurred by a lack of supply satisfaction. Interestingly, in all the cases we have analyzed, the system improvement obtained by considering these costs in the fitness function implies only a moderate increase in the initial investment costs. Of course, other proposals can be found in the literature.

Following (Martínez, 2007) reliability is added from an economic point of view, by considering the costs of the water not delivered due to problems in the system. As a result, the fitness function adds this additional cost:

$$F(D)=\sum_{i=1}^{L} c_i(D_i)L_i+\sum_{j=1}^{N}\mathrm{H}(H_{\min}-H_i)\cdot p\cdot(H_{\min}-H_i)+\sum_{i=1}^{L} w_i\cdot L_i\cdot D_i^{-u}\,. \tag{4}$$

Here, w_i is a coefficient associated with each pipe, in the form $a\cdot t_f\cdot(c_f+c_a\cdot V_f)$; $a\cdot L\cdot D^{-u}$ gives the number of expected failures per year of one pipe, as a function of diameter, D_i, and length, L_i, (a and u are known constants); t_f is the average number of days required to repair the pipe; c_f is the average daily repair cost; c_a is the average cost of the water supplied to affected consumers, in monetary units per unit volume; and $V_f=86400\cdot Q_{\text{break}}$ is the daily volume of water that should be supplied to the affected consumer due to the loss of water of Q_{break} in cubic meters per second.

The scenarios considered here follow the approach of 'breaking', in turn, all the pipes of a specific design to check if all the constraints are fulfilled by the design. If the test is negative, the design is suitably penalized. In this way, designs develop increasing reliability. To undergo these tests, the system must be analyzed for any of the specific 'breakages'.

In the case study we present here – which corresponds to a real-world network – (see Fig. 2) the minimum pressure allowed is 15m, and the available commercial diameters are given in Table 1.

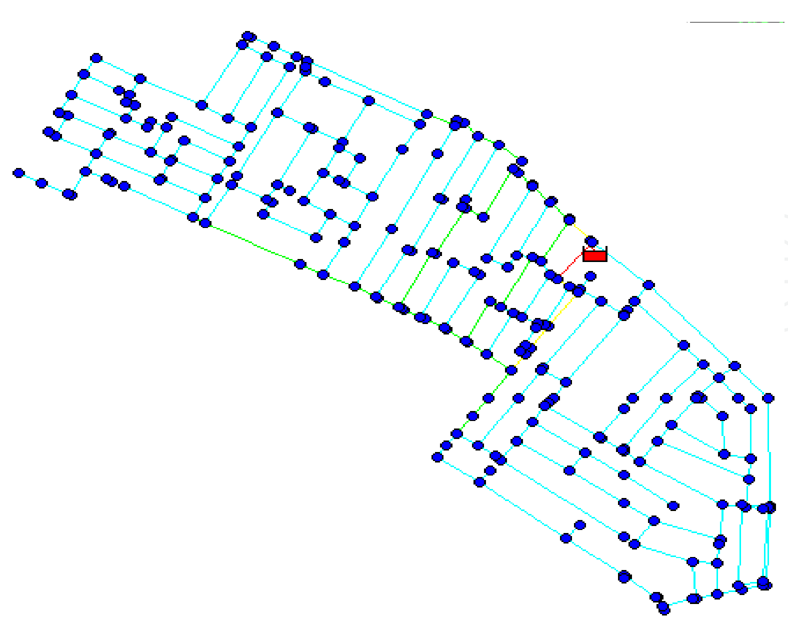


Fig. 2. Layout of the studied network

This table also includes the Hazen-Williams coefficient, *C*, used in the hydraulic model, and the unit cost of the available and variously sized pipes.

This network, which is fed by a tank, has 294 lines amounting to 18.337km of pipes, and 240 nodes consuming 81.53l/s in total. The dimensionality of this problem, which is of moderate size, is immense.

Diameter (mm)	$C_{H\text{-}W}$	Cost ($units)
100	140	117.14
150	140	145.16
200	140	191.42
250	140	241.09
300	140	333.16

Table 1. Commercially available diameters for the case-study in Fig. 2

3.2 The design of WWS

The design of wastewater collection networks involves the simultaneous use of continuous and discrete variables. For the sake of simplicity and comparison, let us consider a network with no special elements, such as pumps, drops, tanks, and other sewer system elements. The decision variables will then be pipe diameters and slopes. While slopes are clearly continuous variables, diameters must be treated as discrete, since they have to adjust to the range of commercial pipes that are available for the design. The PSO version we present in Section 4 is able to deal simultaneously with both continuous and discrete variables. To evaluate the proposed algorithm, a small water collection system has been designed – using both continuous and discrete variables. The network is shown in Fig. 3.

Peak flows, in litres per second, at the inlets of the network are associated with the nodes, numbered 1 to 7. A more general treatment of this problem should consider sets of hydrographs at system inlet points, rather than fixed pipe design flows. As we intend to use dynamic programming for comparison purposes, we only consider steady state behaviour. Nevertheless, various distributions of input flows, enabling more realistic designs that are compatible with different loads, may be straightforwardly considered by the PSO-based technique. In Fig. 3, the line lengths and ground elevation of the nodes are given.

The design constraints consider the ratio of flow depth for the diameter of any pipe to be lower than 80%, and the velocity not to exceed 4m/s. In addition, all the pipes must be buried at least one metre deep. The fitness function accounts for the costs of the pipes and excavation works. Finally, as in any hydraulic problem, continuity and energy equations complete the set of constraints (see Izquierdo et al., 2004). Non-linearity is associated to the energy equations, all the other constraints and fitness function being linear.

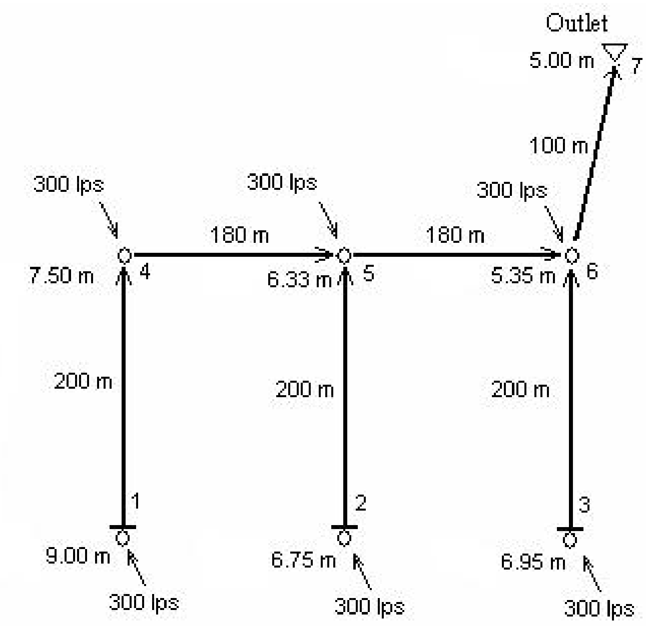


Fig. 3. A sample network

The total cost of the network is considered as the sum of the network cost and a penalty cost, defined as

$$F = \sum_{i=1}^{\#pipes} C_i \cdot L_i + \sum_{i=1}^{\#pipes} Ex_i + \sum_{j=1}^{\#constr} p_j \sum_{i=1}^{\#pipes} v_{ij}^2 \, , \tag{5}$$

where C_i = diameter cost, L_i = length, and Ex_i = excavation costs for pipe i; v_{ij} is the j-th constraint violation at pipe i, and p_j represents the penalty parameter corresponding to constraint j with a large value – to ensure that infeasible solutions have a cost greater than any feasible solution.

3.3 Calibration of a WDS and leak identification

Computational modelling of WDS has become important for WDS authorities. Nevertheless, hydraulic models can only offer a very accurate description of processes provided there is no missing, or uncertain, data; and provided that initial and/or boundary conditions and the forcing terms are precisely defined. In real-life applications, it is a very difficult task to create precise conditions for models. To be useful, any WDS hydraulic model should first be calibrated. Calibration is the process by which a certain number of model parameters are adjusted until the model closely mimics the behaviour of the real system. For WDSs, pipe roughness and leaks are the classes of parameters where uncertainty concentrates. For new pipes, roughness is assessed directly by lab tests. However, for an already existing WDS, the old manual methods used formerly (and even currently) are inaccurate. Much better results can be achieved if calibration of the analyzed WDS model is formulated via optimization problems using a fitness function that tries to reconcile the difference between the measured pressures and the predicted pressures that the hydraulic simulation computer model yields with assumed system parameters. This is a typical example of so-called inverse analysis.

To obtain good values for the system losses, roughness coefficients and flowrates through the pipes must be known with accuracy. Nevertheless, flowrates cannot be accurately assessed if there are any leaks in the system. Leak identification is a difficult problem that can often only be assessed globally through lumped audits performed in certain zones of the network, or in its entirety. There are a number of proposals in the literature for detecting and identifying anomalies and topological errors (see, for example, (Izquierdo et al., 2007), and quotes therein); yet, this state estimation process still represents an important challenge facing water supply managers. For the state estimation process, use is made of the mathematical model of the network. The nonlinear relations among flowrates and heads describing the system balances are confronted by the specific measurements taken using telemetry and other systems. These measurements are compared with the corresponding measurements provided by the model. Discrepancies are then minimized, so enabling roughness coefficients and leak magnitudes to become the variables of the fitness function. The complexity of water systems only enables a few real-time measurements to be made, which only incompletely represent the network state. Obviously, the larger the number of real measurements, the better the accuracy with which roughness coefficients and leaks are obtained. Discrepancies between measured and theoretical (given by the model) pressure heads are then minimized using the fitness function below:

$$F = \sum_{j=1}^{n} p_j \cdot \left| H_j^{\mathrm{m}} - H_j^{\mathrm{c}} \right|, \tag{6}$$

where n is the number of demand junctions where measurements are available, and p_j is the penalty for the discrepancy between H_j^{m}, which is the measured piezometric head at node j, and H_j^{c}, which is the calculated piezometric head at node j. The penalty factor is taken as a large number if $\left| H_j^{\mathrm{m}} - H_j^{\mathrm{c}} \right|$ is larger than a tolerance threshold allowed for node j, and zero otherwise.

By minimizing (6), the problem variables should approach the values of their corresponding real parameters.

4. PSO and the considered variant

All evolutionary algorithms share two prominent features. Firstly, they are population-based. A certain number of individuals, grouped as a population, are used to explore the solution space and find the optimum in the system. In PSO, each bird of the flock is a potential solution and is referred to as a particle. Secondly, there is communication and information exchange among individuals in the test population. In this framework, the birds, besides having individual intelligence, also develop some social behaviour and coordinate their movement towards a destination (Kennedy & Eberhart, 1995). Initially, the process starts from a swarm of M particles, X_i, moving within the search space, $S \subset R^d$, where d is the number of variables involved, each representing a potential solution of the problem

$$\text{Find } \min_{X \in S} F(X) \text{, subject to appropriate constraints,}$$

where F is the fitness function associated with the problem, which we consider to be a minimization problem without loss of generality. The optimal solution is then searched for by iteration. The performance of each particle is measured using this fitness function, according to the problem in hand.

In each cycle of the iteration, t, the i-th particle is associated with the following: (i) its current position, $X_i = (x_{i1}\dots\ x_{id})$; (ii) its best position, $Y_i = (y_{i1}\dots\ y_{id}) = \mathrm{argmin}(F(X_i(t)),\ F(X_i(t\text{-}1)))$, reached in previous cycles; and (iii) its flight velocity $V_i = (v_{i1}\dots\ v_{id})$, which makes it evolve. The bird which is in the best position, $Y^* = \mathrm{argmin}\{F(X_i(t),\ i = 1,\dots,\ M\}$, is identified for every t.

4.1 Manipulation of particles

In each generation, the velocity of each particle is updated – based on its best encountered position, the best position encountered by any particle, and a number of parameters:

$$V_i \leftarrow \omega V_i + c_1\ rand(\ )\ (Y_i - X_i) + c_2\ rand(\ )\ (Y^* - X_i). \tag{7}$$

In each dimension, particle velocities are clamped to minimum and maximum velocities, which are user-defined parameters,

$$V_{\min} \leq V_{ij} \leq V_{\max}, \tag{8}$$

in order to control excessive roaming by particles outside the search space. These very important parameters are problem-dependent. They determine the resolution with which regions between the present position and the target (best so far) positions are searched. If velocities are too great, particles might fly through good solutions. If they are too slow, on the other hand, particles may not explore sufficiently beyond locally good regions – becoming easily trapped in local optima and unable to move far enough to reach a better position in the problem space. Usually, $V_{\min}$ is taken as $-V_{\max}$.

The position of each particle is also updated every generation. This is performed by adding the velocity vector to the position vector,

$$X_i \leftarrow X_i + V_i. \tag{9}$$

The parameters in (7) are as follows: c_1 and c_2 are two positive acceleration constants, called the cognitive and social parameters, respectively; $rand(\ )$ represents a function that creates random numbers between 0 and 1 (two independent random numbers enter Equation (7)); ω is a factor of inertia suggested by (Shi & Eberhart, 1998) that controls the impact of the velocity history on the new velocity.

Expression (7) is used to calculate the i-th particle's new velocity, a determination that takes into consideration three main terms: the particle's previous velocity, the distance of the particle's current position from its own best position, and the distance of the particle's current position from the swarm's best experience (position of the best particle). Thus, each particle or potential solution moves to a new position according to expression (9).

The previously described algorithm can be considered as the standard PSO algorithm, which is applicable to continuous systems and cannot be used for discrete problems. Various approaches have been put forward to tackle discrete problems with PSO (Al-Kazemi & Mohan, 2002; Rastegar et al., 2004; Liao et al., 2007; Shi et al., 2007). The approach we propose for discrete variables plainly involves the use of the integer part of the discrete velocity components. This way, the new velocity of discrete components will be an integer and, as a consequence, the new updated positions will share this characteristic since the initial population, in its turn, must also have been generated using only integer numbers. According to this simple idea, expression (7) will be replaced by

$$V_i \leftarrow fix(\omega V_i + c_1\, rand(\,)\,(Y_i - X_i) + c_2\, rand(\,)\,(Y^* - X_i)), \tag{10}$$

for discrete variables, where $fix(\cdot)$ is a function that takes the integer part of its argument. However, it should be taken into account that the new velocity discrete values must be controlled by suitable bounds as in (8). There is, however, a singular aspect regarding velocity bounds that must be taken into consideration so that the algorithm can treat both continuous and discrete variables in a balanced way. In (Izquierdo et al., 2008a), it was found that using different velocity limits for discrete and continuous variables produces better results.

4.2 Manipulation of parameters

The role of the inertia, ω, in (7) and (10) is considered critical for the convergence behaviour of the PSO algorithm. Although inertia was constant in the early stages of the algorithm, currently it is allowed to vary from one cycle to the next. As it facilitates the balancing of global and local searches, it has been suggested that ω could be allowed to adaptively decrease linearly with time – usually in a way that initially emphasizes global search and then, with each cycle of the iteration, increasingly prioritizes local searches (Shi & Eberhart, 1999). A significant improvement in the performance of PSO, with decreasing inertia weight across the generations, is achieved by using (Jin et al., 2007)

$$\omega = 0.5 + \frac{1}{2(\ln(t)+1)}. \tag{11}$$

However, in the variant we propose, the acceleration coefficients and clamping velocity are neither set to a constant value, as in standard PSO, nor set as a time-varying function, as in adaptive PSO variants (Arumugam & Rao, 2008). Instead, they are incorporated into the optimization problem (Montalvo et al., 2009). Each particle is allowed to self-adaptively set its own parameters by using the same process used by PSO – and given by expressions (7) or (10), and (9). To this end, these three parameters are considered as three new variables that are incorporated into position vectors X_i. In general, if d is the dimension of the problem, and p is the number of self-adapting parameters, the new position vector for particle i will be:

$$X_i = (x_{i1}, \ldots, x_{id}, x_{id+1}, \ldots, x_{id+p}). \tag{12}$$

Obviously, these new variables do not enter the fitness function, but rather they are manipulated by using the same mixed individual-social learning paradigm used in PSO.

Also, V_i and Y_i, which give the velocity and thus-far best position for particle i, increase their dimension, correspondingly.

By using expressions (7) or (10), and (9), each particle is additionally endowed with the ability to adjust its parameters by taking into account the parameters it had at its best position in the past; as well as the parameters of the leader, which facilitated this best-particle's move to its privileged position. As a consequence, particles use their cognition of individual thinking and social cooperation to improve their positions; as well as improving the way they improve their position by accommodating themselves to the best-known conditions, namely, their conditions and their leader's conditions when they achieved the thus-far best position.

4.3 Enriched diversity

PSO's main drawback is the difficulty in maintaining acceptable levels of population diversity while balancing local and global searches; as a result, suboptimal solutions are prematurely obtained (Dong et al., 2005). Some evolutionary techniques maintain population diversity by using some more or less sophisticated operators or parameters. Several other mechanisms for forcing diversity in PSO can be found in the literature (Angeline, 1998; Løvbjerg et al., 2001; Parsopoulos et al., 2001a; Parsopoulos et al., 2001b; Zhang & Xie, 2003). In general, the random character that is typical of evolutionary algorithms adds a degree of diversity to the manipulated populations. Nevertheless, in PSO those random components are unable to add sufficient diversity. As shown in (Montalvo et al., 2008b) frequent collisions of birds in the search space, especially with the leader, can be detected. This caused the effective size of the population to fall and the algorithm's effectiveness to be consequently impaired. The study in (Izquierdo et al., 2009a) introduces a PSO derivative in which a few of the best birds are selected to check collisions, and colliding birds are randomly re-generated if collision occurs. This random re-generation of the many birds that tend to collide with the best birds has been shown to avoid premature convergence as it prevents clone populations from dominating the search. The inclusion of this procedure into PSO greatly increases diversity; as well as improving convergence characteristics and the quality of the final solutions.

4.4 The algorithm

The modified algorithm can be given by the following pseudo-code, with t as the iteration number.

- $t = 0$
- Generate a random population of M particles: $\{X_i(t)\}_{i=1}^{M}$, according to (12)
- Evaluate the fitness of the particles (only the first d variables enter the fitness function)
- Record the local best locations $\{Y_i(t)\}_{i=1}^{M}$; the values of the corresponding parameters are also recorded
- Record the global best location, $Y^*(t)$, and the list of the m best particles to check collisions (including their corresponding parameters)
- While (not termination-condition) perform the following:

  - Determine the inertia parameter $\omega(t)$, according to (11)
  - Begin cycle from 1 to number of particles M
    - Start
      - Calculate new velocity, $V_i(t+1)$, for particle i according to (7), and take its integer part (for discrete optimization) for the first d variables, according to (10)
      - Update position, $X_i(t+1)$, of particle i according to (9)
      - Calculate fitness function for particle i
      - If particle i has better fitness value than the fitness value of the best particle in history, then set particle i as the new best particle in history, and update the list of the m best particles
      - If particle i is not currently one of the m best particles but coincides with one of the selected m best particles, then re-generate particle i randomly (including its parameters)
    - End
  - $t = t + 1$
- Show the solution given by the best particle

Different termination conditions, such as the number of fitness function evaluations, maximum run time, convergence in the fitness or search space, may be stated (Shi et al., 2007). For most of the cases considered here, the termination condition stops the process if, after a pre-fixed number of iterations, no improvement in the solution had been obtained.

The performance of the approach introduced herein can be observed in the next section – using the results obtained for the problems presented in Section 3.

5. Application of the PSO considered variant to the addressed problems

5.1 Design of WDSs

In (Izquierdo et al., 2009a) it is shown that, for the NYT and Hanoi networks, a small representative sample of the algorithm’s runs can be used to consistently achieve near optimal results at a much reduced computational cost, which is of paramount importance from a practical point of view. By using the obtained results, the probability of a single run obtaining a solution differing by less than a certain percentage from the best-known solution was obtained. These probabilities have been plotted in Figure 4. It can be observed that for both studied problems, for example, one single run of our algorithm gives a solution that is less expensive than 5.5% of the best-known solutions with a probability of 86%. Additionally, there is an almost complete guarantee that in only one single run of the algorithm, a solution will be obtained with a cost under 1.1 times the best-known solution cost.

The best solution found by the used PSO variant is shown in Table 2 for the New York system, and in Table 3 for the Hanoi system; together with other best solutions found in the literature.

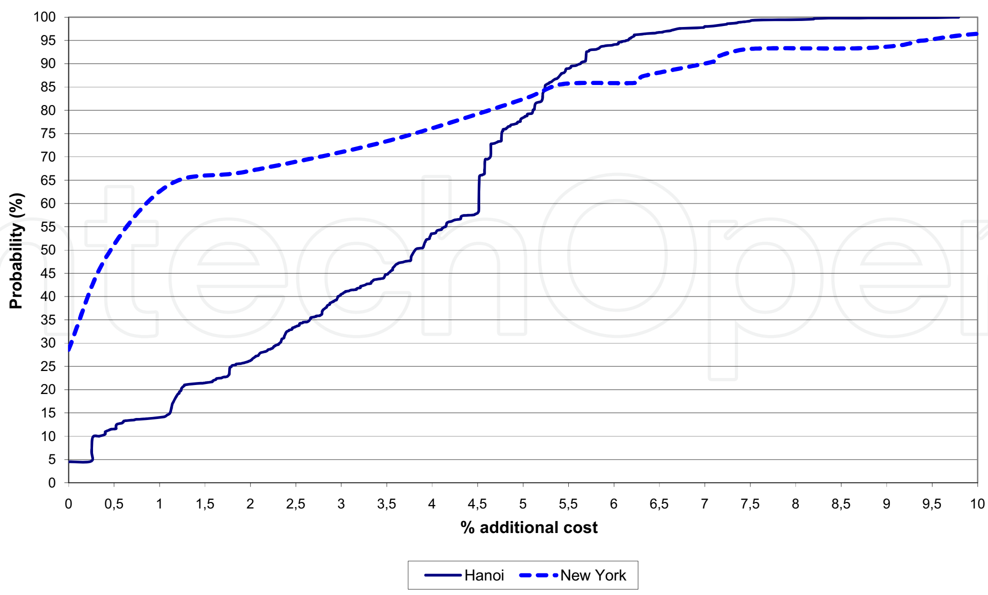


Fig. 4. Probability of first-run good solution

Method Ref.	GA (Matías, 2003)	GA (Zecchin, 2003)	ACO (Maier et al. 2003)	GA (Savic & Walters, 1997)	PSO (Montalvo et al., 2008a)	PSO (Montalvo et al., 2009)
Cost	38.64	38.8	38.64	40.42	38.64	38.64

Table 2. Optimal design cost (×$\$10^6$) for the NYT network

Method Ref.	GA (Matías, 2003)	GA (Wu & Simpson, 2001)	GA (Savic & Walters, 1997)	ACO (Zecchin et al., 2005)	PSO (Montalvo et al., 2008a)	PSO (Montalvo et al., 2009)
Cost	6.093	6.182	6.195	6.367	6.133	6.081

Table 3. Optimal design cost (×$\$10^6$) for the Hanoi network

Parameter V_{max} was set to 50% of maximum variable range, and the stopping condition to 200 iterations. The number of used particles was 100.

The real-world problem in Figure 2 is solved (Izquierdo et al., 2009b) by using the two fitness functions defined in (4) and (2), with and without reliability considerations, respectively. A colour code has been used to aid an understanding of the results. With reference to the pipes: the blue, green, yellow, and red colours represent 100, 150, 200, and 250mm pipes, respectively. Regarding nodes: dark blue represents pressure above 15m;

light blue, between 14 and 15m; green, between 12 and 14m; yellow, between 10 and 12m; and, finally, nodes with a pressure under 10m are represented in red.

This network, which is fed by a tank, has 294 lines amounting to 18.337km of pipes and 240 nodes consuming 81.53l/s in total. Figure 3 (left) presents the solution obtained by using (4), which includes reliability. This solution is a mere 3.65% more expensive than the solution obtained using (2), with no reliability consideration. The diameters for this last case can be observed in Figure 3 (right). Table 4 presents a comparison of the initial investment costs for both solutions.

Diameter (mm)	Without reliability		With reliability	
	Length (m)	Cost ($units)	Length (m)	Cost ($units)
100	17 731.10	2 077 021.41	15 822.31	1 853 425.63
150	606.39	88 023.28	2077.69	301 597.04
200	0.00	0.00	328.79	62 937.56
250	0.00	0.00	108.70	26 206.24
300	0.00	0.00	0.00	0.00
Total cost ($units)	2 165 044.69		2 244 166.47	

Table 4. Comparison between costs for both solutions

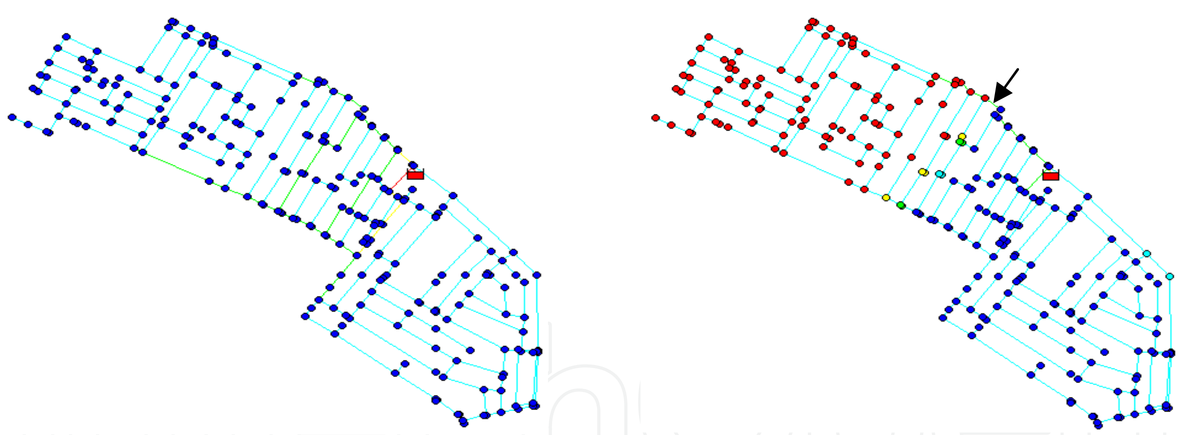


Fig. 3. Network design and state: (left) with reliability no matter what pipe is out of service; (right) without reliability with the marked pipe out of service

The effect of closing the pipe indicated by the arrow can be observed in Figure 3 (right) for the solution without reliability. It shows the considerable impact produced by a closed pipe: almost half of the nodes (those not in dark blue) do not have the minimum required pressure of 15m. Figure 4 (left) shows that this does not happen for the more reliable design obtained by using (4) since, no matter which pipe is out of service, all the nodes are in dark blue.

5.2 The design of WWS

Flow behaviour through the pipes can be modelled by any of the codes to analyze sewer systems available in the market. In this particular case, we have used the EPA-SWMM package (Rossman, 2005). Even though any part of the analysis provided by this package may be used within the evolutionary algorithm, in the case of this paper only steady state formulation (for peak flows) will be performed for comparison purposes. In effect, given the low dimensionality of the network under study, good results can be obtained using dynamic programming. We use these results as a reference to compare with the results given by PSO, since dynamic programming is theoretically capable of finding the global optimum solution with the only limitation of the discretization used for continuous variables (pipe slopes in our case).

In this case, parameter V_{max} was set to 50% of maximum variable range for diameters, and 20% of variable range for slopes (continuous variables); and the stopping condition was set at 800 iterations.

Sixty executions were performed for the problem at hand. In only one of the sixty cases was the algorithm unable to find a feasible solution. In the other 98.3% of cases, different feasible solutions were found. The average cost for these best solutions is 221.29×€10^3. The best solution is 203.055×€10^3. Also, the same network was designed by using dynamic programming, giving a cost of 206.7×€10^3. Therefore, PSO has found a better solution than the solution provided by dynamic programming, since a discretization of 0.2m for the excavation depth has been used with this last technique. However, an additional experiment using dynamic programming with a finer discretization of 0.1m was also performed, and the obtained result was 204.0×€10^3. Of course, this value is lower than the value obtained with the coarser discretization, but it still does not improve upon the best solution obtained with PSO. Thus, this algorithm shows itself able to go beyond the limits of very fine dynamic programming discretizations.

5.3 Calibration of WDSs and leak identification

The proposed procedure has been applied to the Hanoi network, already considered in section 3.1. The network topology uses the design data given in (Wu & Simpson, 2001) regarding the length of the pipes and the demand at the junctions. As stated by (Wu & Simpson, 2001), pipe diameters were unknown, since it was a design problem. Here, we have assigned design values to the diameters of the pipes. Also a roughness coefficient of 130 (*C* of Hazen-Williams) has been assigned to all the pipes. Additionally, five new demand nodes have been considered in order to mimic the system leaks. By using EPANET2 (Rossman, 2000), the network was analyzed and the computed head values at the junctions were stored. These pressure heads, together with the assigned Hazen-Williams coefficient and the localization and magnitude of the leaks, synthetically represent the real (measured) values of the network.

To assess the performance of the algorithm, roughness coefficients were allowed to vary between 80 and 140, and leak exponents between 0 and 1.5 for the original network with identification of leaking pipes.

The algorithm was run 100 times, and always the difference between measured and calculated pressure heads was smaller than 0.15mca.

Parameter V_{max} was set to 7% of maximum variable range. The execution stopped if no improvement was obtained after 200 iterations, or if the fitness function reached a value of 0. A population of 300 particles was used.

6. Conclusion

Optimization problems in the field of urban hydraulics are complex by nature and difficult to solve using conventional optimization techniques. In particular, for large WDSs and WWSs, the optimization process for construction and maintenance requires considerable resources every year. Also, growing concern has arisen over water loss in existing WDSs, as they often contain many aging elements, and so the calibration of friction and leakage is of paramount importance in urban water systems. These problems can involve both continuous and discrete variables. PSO is an evolutionary optimization algorithm that can be adapted to deal with both types of variables.

In this work, PSO has been applied to three different problems and good solutions have been found in all the considered case-studies. The ability to deal with both continuous and discrete variables, as well as a feature that increases the diversity of the population of birds, has been considered. This feature makes the algorithm converge with less iteration, thus saving time, which is of paramount importance in real problems related to water systems. Finally, the laborious aspect that pervades all metaheuristics regarding how to perform appropriate parameter adjustments has been overcome by using a self-adaptive parameter control, which renders PSO parameters subject to evolution. Additionally, this formulation obviates the tedious pre-processing task of parameter fine-tuning.

The first problem is the design of WDSs. The performance of the considered PSO variant has been illustrated by application to two well-known benchmark networks, and the results have been compared with those obtained using other evolutionary algorithms. Comparison of the results shows that this formulation finds optimum, or near-optimum, solutions much more efficiently, and with considerably less computational effort. It is noteworthy that for the Hanoi system, the average cost of the 100 performed runs was 6.297 million dollars, only 3.56% higher than the best-known solution. In the case of the New York system, the result is 39.738 million dollars or 2.91% higher than the best-known solution. The average number of generations needed to obtain the best solution for the Hanoi system is 700, with 105 being the minimum number of generations to obtain the best solution. For the New York system, these figures are 230 generations for the best solutions, and 16 for the minimum number of generations to obtain the best solution.

We also have tackled the robust design of such a WDS. The solution cannot ignore the evaluation of aspects related to different scenarios and certain failure conditions. By considering only the initial investment costs, cheaper designs will be produced; but these designs will suffer serious difficulties coping with abnormal situations. We have shown, through one real-world case study, that more reliable designs do not necessarily involve immoderate increases in investment. Interestingly, the same case study shows, nonetheless, much better performances for reliable designs when failure events are represented by pipes being out of service. The concept of reliability used here takes into account the economic impact of the water not delivered due to this type of failure during the life of the network.

In the problem of the design of wastewater systems, PSO performance was compared with an exact method, namely dynamic programming. PSO found a better solution than that provided by dynamic programming when a discretization of 0.2m for the excavation depth was used. However, after refining this discretization to 0.1m, PSO still outperforms dynamic programming. Therefore, PSO is able to go further beyond the limits of very fine dynamic programming discretizations, and interestingly, avoids becoming trapped in the dimensionality curse.

This algorithm has also shown good performance when applied to the problem of calibration and leak identification in a WDS. Pressure differences were lower than 0.15m in all the 100 algorithm executions for the considered case-study – after having found suitable values for pipe roughness and leak magnitudes. New studies should be performed with reduced redundancy in the number of measured parameters. The same approach should be also applied to other networks, since this is a problem that has received little attention in the literature.

The main advantages of the method are that it does not require sophisticated operators, nor parameters: and is thus simpler than other evolutionary techniques. This method does not need initial feasible particles, nor do the re-generated particles need to be feasible; and finally, it is robust in handling diverse fitness functions and various constraints. Furthermore, having fewer generations is a major advantage in real water distribution systems, where cost and time constraints prohibit repeated runs of an algorithm and hydraulic evaluations. From the studied benchmark problems, it can be inferred that obtaining 'good' solutions with the proposed algorithm is straightforward, since there is no need for *a priori* parametric study. This algorithm is also relatively inexpensive as the computational cost increased only slightly because of the relatively small increase in the dimensionality of the search space. Therefore, the algorithm is desirable when the goal is to quickly obtain good solutions that are not necessarily very close to the optimum.

Optimization has been carried out using a variant of PSO, devised by the authors, that considers both discrete and continuous variables. This variant has increased population diversity and self-adaptively manages its parameters. This optimization tool is very useful since other terms (load or service conditions, rehabilitation costs, life-long costs, other reliability measurements, and so on) can be added to the fitness function without rendering the problem more conceptually complex. In addition, this tool can be easily combined with hydraulic network simulation modules, so allowing great versatility in the analysis of candidate solutions.

Finally, the abilities of these particles to decide, as a group, how to move inside the search space, and change their behaviour during the search processes, as well as finding very good solutions in a relatively short period of time, constitutes an open-door environment that could be perfectly exploited to address multi-objective formulations regarding optimization problems in different fields. If one of these capabilities is missing, the PSO algorithm would not be so useful as an optimization tool; and it is precisely the assembly of these capabilities that makes the PSO algorithm a powerful multi-agent system for solving problems in the water industry.

7. References

Al-Kazemi, B. & Mohan, C.K. (2002). Multi-phase discrete particle swarm optimization, *JCIS*, pp. 622–625.

Angeline, P.J. (1998). Using selection to improve particle swarm optimization, *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 84–89, Anchorage, Alaska, USA.

Arumugam, M.S. & Rao, M.V.C. (2008). On the improved performances of the particle swarm optimization algorithms with adaptive parameters, cross-over operators and root mean square (RMS) variants for computing optimal control of a class of hybrid systems. *Appl. Soft Comput.*, 8(1): 324-336.

Botrous, A.; El-Hattab, I. & Dahab, M. (2000). Design of wastewater collection networks using dynamic programming optimization technique, *Proceedings of the ASCE Nat. Conf. on Environmental and pipeline Engineering*, pp. 503-512, Kansas City, MO, United States, American Society of Civil Engineers.

Cunha, M.C. & Sousa, J. (1999). Water distribution network design optimization: simulated annealing approach, *Journal of Water Resources Planning and Management*, 125 (4), 214–221.

Dandy, G.C.; Simpson, A.R. & Murphy, L.J. (1996). An improved genetic algorithm for pipe network optimization, *Water Resources Research*, 32(2), 449-458.

Dong, Y., Tang, B.X.J. & Wang, D. (2005). An application of swarm optimization to nonlinear programming, *Computers & Mathematics with Applications* 49 (11-12), pp. 1655–1668.

Geem, Z.W. (2006). Optimal cost design of water distribution networks using harmony search, *Engineering Optimization*, 38 (3), pp. 259–280.

Goulter, I.C. & Coals, A.V. (1986). Quantitative approaches to reliability assessment in pipe networks, *Journal of Transportation Engineering*, 112 (3), 287–301.

Goulter, I.C. & Bouchart, F. (1990). Reliability-constrained pipe network model, *Journal of Hydraulic Engineering*, 116 (2), 211–229.

Herrera, M.; Izquierdo, J.; Montalvo, I.; García-Armengol, J. & Roig, J.V. (2009). Identification of surgical practice patterns using evolutionary cluster analysis, *Mathematical and Computer Modelling,* doi: 10.1016/j.mcm.2008.12.026.

Izquierdo, J.; Pérez, R. & Iglesias, P.L. (2004). Mathematical models and methods in the water industry, *Mathematical and Computer Modelling*, 39 (11-12), 1353–1374.

Izquierdo, J.; López, P.A.; Martínez, F.J. & Pérez, R. (2007). Fault detection in water supply systems using hybrid (theory and data-driven) modelling, *Mathematical and Computer Modelling*, 46, 3-4, 341-350.

Izquierdo, J.; Montalvo, I.; Pérez, R. & Fuertes, V.S. (2008a). Design optimization of wastewater collection networks by PSO, *Computer & Mathematics with Applications*, 56(3), 777–784.

Izquierdo, J.; Minciardi, R.; Montalvo, I.; Robba, M. & Tavera, M. (2008b). Particle Swarm Optimization for the biomass supply chain strategic planning, *Proceedings of 4th Biennal Meeting, iEMSs 2008: International Congress on Environmental Modelling and Software*, pp. 1272-1280, Barcelona, Spain.

Izquierdo, J.; Montalvo, I.; Herrera, M. & Pérez, R. (2009a). A derivative of Particle Swarm Optimization with enriched diversity, submitted to Computational Optimization and Applications.

Izquierdo, J.; Montalvo, I.; Pérez, R. & Herrera, M. (2009b). Robust Design of Water Supply Systems through Evolutionary Optimization. *POSTA09. Third Multidisciplinary Symposium on Positive Systems: Theory and Applications*, accepted, Valencia, Spain.

Janson, S.; Merkle, D. & Middendorf, M. (2008). Molecular docking with multi-objective Particle Swarm Optimization, *Applied Soft Computing* 8, 666–675.

Jin, Y.X.; Cheng, H.Z.; Yan, J.I. & Zhang, L. (2007). New discrete method for particle swarm optimization and its application in transmission network expansion planning, *Electric Power Systems Research*, 77(3-4), 227-233.

Kennedy, J. & Eberhart, R.C. (1995). Particle swarm optimization, *Proceedings of the IEEE International Conf. on Neural Networks*, pp. 1942-1948, Piscataway, NJ, USA.

Liao, C.J.; Tseng, C.T. & Luarn, P. (2007). A discrete version of particle swarm optimization for flowshop scheduling problems, *Comput. Oper. Res.*, 34 (10), 3099–3111.

Liong, S.Y. & Atiquzzaman, M. (2004). Optimal design of water distribution network using shuffled complex evolution, *Journal of the Institution of Engineers*, Singapore 44 (1), 93–107.

Løvbjerg, M.; Rasmussen, T.K. & Krink, T. (2001). Hybrid particle swarm optimiser with breeding and subpopulations, *Proceedings of the Genetic and Evolutionary Computation Conf.*

Maier, H.R.; Simpson, A.R.; Zecchin, A.C.; Foong, W.K.; Phang, K.Y.; Seah, H.Y. & Tan, C.L. (2003). Ant colony optimization for design of water distribution systems, *Journal of Water Resources Planning and Management*, 129 (3), pp. 200–209.

Mariles, Ó.A.F. & Nava, A.P. (2007). Calibración del factor de fricción y localización de fugas en una red de tuberías de agua potable, *Proceedings of the VII SEREA - Seminario Iberoamericano sobre Planificación, Proyecto y Operación de Sistemas de Abastecimiento de Agua*, Morelia, México.

Martínez, J.B. (2007). Quantifying the economy of water supply looped networks. *Journal of Hydraulic Engineering*, ASCE, 133(1): 88-97.

Matías, A.S. (2003). Diseño de redes de distribución de agua contemplando la fiabilidad, mediante algoritmos genéticos, *PhD thesis*, Universidad Politécnica de Valencia, Valencia, Spain.

Montalvo, I.; Izquierdo, J.; Pérez, R. & Tung, M.M. (2008a). Particle Swarm Optimization applied to the design of water supply systems, *Computer & Mathematics with Applications*, 56(3), 769–776.

Montalvo, I.; Izquierdo, J.; Pérez, R. & Iglesias, P.L. (2008b). A diversity-enriched variant of discrete PSO applied to the design of Water Distribution Networks, *Engineering Optimization*, 40(7), 655–668.

Montalvo, I.; Izquierdo, J.; Pérez, R. & Herrera, M. (2009) Improved performance of PSO with self-adaptive parameters for computing the optimal design of Water Supply Systems, submitted to Computer-Aided Design.

Pan, Q.K.; Tasgetiren, F. & Liang, Y.C. (2008). A discrete particle swarm optimization algorithm for the no-wait flowshop scheduling problem, *Computers and Operations Research*, 35(9), 2807-2839.

Parsopoulos, K.E.; Plagianakos, V.P.; Magoulas, G.D. & Vrahatis, M.N. (2001a). Stretching technique for obtaining global minimizers through particle swarm optimization, *Proceedings of the Workshop on Particle Swarm Optimization*, Indianapolis, IN, USA.

Parsopoulos, K.E.; Plagianakos, V.P.; Magoulas, G.D. & Vrahatis, M.N. (2001b). Improving particle swarm optimizer by function stretching, in: *Advances in Convex Analysis and Global Optimization*, Nonconvex Optimization and Applications series, pp. 445–457, Kluwer Academic Publishers, The Netherlands.

Rastegar, R.; Meybodi, M.R. & Badie, K. (2004). A new discrete binary particle swarm optimization based on learning automata, *Proceedings of the 2004 International Conference on Machine Learning and Applications*.

Rossman, L. (2005). *Storm Water Management Model user's manual (version 5.0)*. U.S. Environmental Protection Agency, Cincinnati, USA.

Rossman, L.A. (2000). *EPANET, users manual*, U.S. Environmental Protection Agency, Cincinnati, USA.

Savic, D.A. & Walters, G.A. (1997). Genetic algorithms for least cost design of water distribution networks, *Journal of Water Resources Planning and Management*, 123 (2), 67–77.

Savic, D.A. (2005). Coping with risk and uncertainty in urban water infrastructure rehabilitation planning, *Acqua e città - i convegno nazionale di idraulica urbana*, S'Agnello (NA), pp. 28-30.

Shi, X.H.; Liang, Y.C.; Lee, H.P.; Lu, C. & Wang, Q.X. (2007). Particle swarm optimization-based algorithms for tsp and generalized tsp, *Inf. Process. Lett.*, 103 (5), pp. 169–176.

Shi, Y. & Eberhart, R.C. (1998). A modified particle swarm optimizer, *Proceedings of the IEEE Congress on Evolutionary Computation*, pp. 69-73, Piscataway, NJ, USA.

Shi, Y & Eberhart, R.C. (1999). Empirical study of particle swarm optimization. *Proceedings of the IEEE Congress on Evolutionary Computation*. Washington, DC, USA.

Wu, Z.Y. & Simpson, A.R. (2001). Competent genetic-evolutionary optimization of water distribution systems, *Journal of Computing in Civil Engineering*, 15 (2), 89–101.

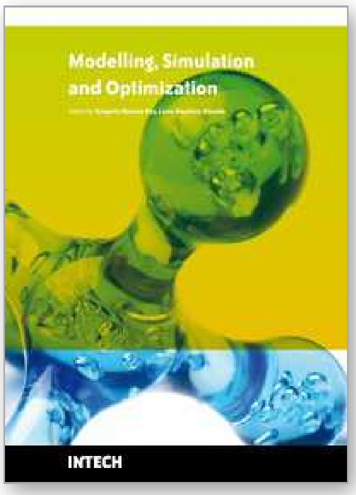
Wu, Z.Y. & Walski, T. (2005). Self-adaptive penalty cost for optimal design of water distribution systems, *Journal of Water Resources Planning and Management*, 131 (3), 181–192.

Zecchin, A. (2003). Max-min ant system applied to water distribution system optimisation, in: *MODSIM 2003: International Congress on Modelling and Simulation*.

Zecchin, A.C.; Simpson, A.R.; Maier, H.R. & Nixon, J.B. (2005). Parametric study for an ant algorithm applied to water distribution system optimization, *IEEE Trans. Evolutionary Computation*, 9 (2), 175–191.

Zecchin, A.C.; Simpson, A.R.; Mayer, H.R.; Leonard, M.; Roberts, A.J. & Berrisford, M.J. (2006). Application of two ant colony optimisation algorithms to water distribution system optimisation, *Mathematical and Computer Modelling*, 44(5-6), pp. 451-468.

Zhang, W.J. & Xie, X.F. (2003). DEPSO: hybrid particle swarm with differential evolution operator, in: *IEEE International Conf. on Systems, Man and Cybernetics*, 2003, vol. 4, pp. 3816–3821.



Modelling Simulation and Optimization

Edited by Gregorio Romero Rey and Luisa Martinez Muneta

ISBN 978-953-307-048-3
Hard cover, 708 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

Computer-Aided Design and system analysis aim to find mathematical models that allow emulating the behaviour of components and facilities. The high competitiveness in industry, the little time available for product development and the high cost in terms of time and money of producing the initial prototypes means that the computer-aided design and analysis of products are taking on major importance. On the other hand, in most areas of engineering the components of a system are interconnected and belong to different domains of physics (mechanics, electrics, hydraulics, thermal...). When developing a complete multidisciplinary system, it needs to integrate a design procedure to ensure that it will be successfully achieved. Engineering systems require an analysis of their dynamic behaviour (evolution over time or path of their different variables). The purpose of modelling and simulating dynamic systems is to generate a set of algebraic and differential equations or a mathematical model. In order to perform rapid product optimisation iterations, the models must be formulated and evaluated in the most efficient way. Automated environments contribute to this. One of the pioneers of simulation technology in medicine defines simulation as a technique, not a technology, that replaces real experiences with guided experiences reproducing important aspects of the real world in a fully interactive fashion [iii]. In the following chapters the reader will be introduced to the world of simulation in topics of current interest such as medicine, military purposes and their use in industry for diverse applications that range from the use of networks to combining thermal, chemical or electrical aspects, among others. We hope that after reading the different sections of this book we will have succeeded in bringing across what the scientific community is doing in the field of simulation and that it will be to your interest and liking. Lastly, we would like to thank all the authors for their excellent contributions in the different areas of simulation.

Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821