Particle Swarm Optimization for the biomass supply chain strategic planning

Joaquín Izquierdo

R. Minciardi

Idel Montalvo

M. Robba

M. Tavera

iEMSs 2008: International Congress on Environmental Modelling and Software
Integrating Sciences and Information Technology for Environmental Assessment and Decision Making
4th Biennial Meeting of iEMSs, http://www.iemss.org/iemss2008/index.php?n=Main.Proceedings
M. Sànchez-Marrè, J. Béjar, J. Comas, A. Rizzoli and G. Guariso (Eds.)
International Environmental Modelling and Software Society (iEMSs), 2008

Particle Swarm Optimization for the biomass supply chain strategic planning

J. Izquierdo[1], R. Minciardi[2], I. Montalvo[1], M. Robba[2], M. Tavera[1]

[1] *Centro Multidisciplinar de Modelación de Fluidos, Universidad Politécnica de Valencia, Camino de Vera s/n, 46022, Valencia, Spain (jizquier,imontalvo@gmmf.upv.es, mata4@doctor.upv.es)*
[2] *DIST, Department of Communication, Computer and System Sciences, University of Genova, via Opera Pia 13, 16145, Genova, Italy (michela.robba@unige.it)*

Abstract: Particle Swarm Optimization is a very well established evolutionary optimization technique that has shown great potential for the solution of various optimization problems. In this paper, a real-life application, namely a biomass supply chain where optimal biomass flows from sources to energy production plants must be suitably defined, is tackled by using a variant of PSO recently introduced by the authors. In particular, the performance of the variant herein proposed is investigated by applying the model to the strategic planning of the biomass supply chain. The optimization problem is non-linear with binary and continuous decision variables and is based on a previous formalization regarding the biomass strategic planning, but with the addition of integer decision variables for plant technologies selection. The model has been applied to the mountain community of Val Bormida (Savona district, Italy).

Keywords: Biomass supply chain; Strategic planning; Optimization; Evolutionary algorithm; Particle Swarm Optimization.

1. INTRODUCTION

Classical methods of optimization involve the use of gradients or higher-order derivatives of the fitness function. But they are not well suited for many real world problems since they are not able to process inaccurate, noisy, discrete and complex data [Bonabeau et al., 1999; Kennedy and Eberhart, 2001]. Thus, robust methods of optimization are often required to generate suitable results. Recently, a number of algorithms that imitate certain natural principles, evolutionary algorithms like Genetic Algorithms, Ant Colony Optimization, Particle Swarm Optimization, Harmony Search, have provided with robustness to different applications.

Among them, Particle Swarm Optimization (PSO) is a very well established evolutionary optimization technique that has shown great potential and good perspective for the solution of various optimization problems [Dong et al., 2005; Izquierdo et al., 2008; Janson et al., 2008; Jin et al., 2007; Liao et al., 2007; Montalvo et al., 2008a; Pan et al., 2007]. Swarm intelligence is a relatively new category of stochastic, population-based optimization algorithms that are closely related to evolutionary algorithms based on procedures that imitate natural evolution. Swarm intelligence algorithms draw inspiration from the collective behavior and emergent intelligence that arise in socially organized populations.

Originally designed to deal with continuous variables, the PSO derivative we consider here overcomes several typical features of this optimization technique. For one thing, PSO is adapted to consider mixed discrete-continuous optimization since the problem we tackle here involves the use of both continuous and discrete variables [Izquierdo et al., 2008; Montalvo et al., 2008a]. For another, one of the main drawbacks associated with PSO comes from the fact that it is difficult to keep good levels of population diversity and to balance local and global searches. This formulation is able to find optimum or near-optimum solutions much more efficiently and with considerably less computational effort because of the richer population diversity it introduces [Montalvo et al., 2008b; Izquierdo et al., 2007]. Needing a low number of generations is a major advantage in real world problems, where costs and time constraints prohibit repeated runs of the algorithm and function evaluations. Finally, the cumbersome aspect, common to all metaheuristics, of choosing the right parameter values is tackled through self-adaptive dynamic parameter control [Montalvo et al., 2008].

One of its real-life applications is a biomass supply chain where optimal biomass flows from sources to energy production plants must be suitably defined. In particular, the performance of the variant herein proposed is investigated by applying the model to the strategic planning of the biomass supply chain.

The decision variables are represented by plant capacity, biomass yearly harvested in a specific forest parcel, and the presence/absence of a specific technology (pyrolysis, gasification or combustion), while the objective function is the sum of costs related to transportation, plant costs, and biomass collection costs. The localization of the conversion plant is assumed not to be a matter of decision but it is pre-specified by the user.

The optimization problem is non-linear with binary and continuous decision variables. It is based on a previous formalization regarding the biomass strategic planning approach [Freppaz et al., 2004; Frombo et al., 2006; Frombo et al., 2007], but with the addition of binary decision variables for plant technologies selection.

The model has been applied to the mountain community of Val Bormida (Savona district, Italy). In this area, the installation of a biomass-to-energy plant in the Cairo Montenotte municipality has been evaluated. The Val Bormida community is constituted by 18 municipalities for a total area of over 53000 hectares. The tree density index, about 75% (that corresponds to a forest vegetation of about 40000 hectares), is high, even compared with the Italian tree index, that is about 60%. The whole forest territory cannot be used due to legislative constraints (natural parks and protected areas), the existence of areas of bio-naturalistic relevance, and because of the occurrence of forest fires and hydro geological disasters. The whole Val Bormida forest vegetation and the area that is available for exploitation is about 27000 hectares. The available forest area corresponds to 506 forest parcels to be considered in the optimization problem. Each parcel is characterized by one main typology of biomass among four typologies (European Beech, Sweet Chestnut, Mixed and Hardwood Coppice) and by different slope classes.

The remainder of this paper is organized as follows. Next section introduces the optimization problem for the biomass supply chain at a strategic planning. Then, the fitness evaluation and the search space for the mathematical formalization are presented. Section 3 provides the rules for the manipulation of the particles in each iteration and explains how parameters are controlled. Also, the main features of the PSO derivative we consider here are introduced. Finally, the main results are reported. A conclusion section wraps up the paper.

2. THE OPTIMIZATION PROBLEM

The decision problem faced in this work regards the definition of the plant's size and kind for energy production from woody biomasses. Moreover, biomass collection over the territory should be defined, taking into account also slope characteristics. In the following, the decision variables, the objectives, and the constraints of the decision problem are described in detail.

2.1 Decision variables

The decision variables are those quantities that represent the decisions to be taken, and are necessary to formalize the objective function and the constraints. In particular, they are:

- $u_{i,j}$: the annual biomass quantity harvested in the *j*-th slope class, [%], $j = 1,...,J$, of the *i*-th parcel, $i = 1,...,N$, [m^{3}y^{-1}];
- δ_k : a binary variable that states the presence of the technology in a known location.

2.2 The objective function

The objective function takes into account the costs and the benefits of the decision problem. In particular, in order to evaluate the overall cost function to be minimized, it is necessary to consider felling and processing, primary transportation, transportation, purchasing and plant costs, as well as the benefits deriving from the sales of the products. The objective function is then composed by six terms that can be expressed as a function of the decision variables:

- C_{FP} , forest biomasses felling and processing costs. They depend on the required time for the operations (that is, of course, proportional to the harvested biomass and to the operation productivity), and on the number and types of operations executed in the forest;
- C_{FT} , forest biomass primary transportation cost. It corresponds to the transport from the felling areas to the landing points near the first available road;
- C_T , transportation cost from landing points to the plant;
- C_P , non forest biomasses (i.e., agricultural or industrial residues) purchasing cost;
- $C_{I,k}$, plant cost related to the *k*-th plant installation and management;
- B_k , benefits deriving from the products sale for the *k*-th plant.

The objective function to be minimized is then

$$C = C_{FP} + C_{FT} + C_T + C_P + \sum_{k=1}^{K} (C_{I,k}\delta_k - B_k\delta_k) \tag{1}$$

with

$$C_{FP} = \sum_{i=1}^{N}\sum_{j=1}^{5} \tilde{C}^{FP} \frac{u_{i,j}}{Pr^{FP}} \left(1 - \Delta_{Del}\delta_{Del} - \Delta_{Deb}\delta_{Deb} - \Delta_{Cc}\delta_{Cc}\right) \tag{2}$$

$$C_{FT} = \sum_{i=1}^{N}\sum_{j=1}^{J} u_{i,j} \frac{\tilde{C}_j^{FT}}{Pr_{i,j}^{FT}} \tag{3}$$

$$C_T = \sum_{i=1}^{N}\sum_{j=1}^{J} d_i^{LW}\ u_{i,j}\ VM_i\ \tilde{C}^T \tag{4}$$

$$C_P = \sum_{\substack{i=1 \\ i\in\phi_{NF}}}^{N} u_i VM_i \tilde{C}_i^P \tag{5}$$

$$C_{I,k} = \tilde{C}_k^I HVP_k\, ETA_k + \tilde{C}_k^M \sum_{i=1}^{N}\sum_{j=1}^{J} \left(u_{i,j} VM_i\right) \tag{6}$$

$$B_k = Ep_k^{Th} P^{Th} + Ep_k^{El} P^{El} + Ep_k^{Bd} P^{Bd} \tag{7}$$

$$Ep_k^{El} = \delta_{El,k}\left[\ \left(HVP_k\, ETA_k^t h\right) - Dt\ \ \right] ETA_k^e \tag{7a}$$

$$Ep_k^{Th} = \delta_{Th,k}\left[\ \left(HVP_k\, ETA_k^t h\right) - \left(1 - Dt\right)\ \right] \tag{7b}$$

$$Ep_k^{B} = \delta_{Bd,k}\left[\ HVP_k\, ETA_k^b h\ \ \right] \tag{7c}$$

where:

- $\tilde{C}^{FP}$ is the unit cost for the forest felling and processing phase, [€ m^{-3}];
- Pr^{FP} is productivity for the process of felling and processing;
- Δ is necessary time to execute the different operations of felling and processing (delimbing, debarking, and cross cutting);
- δ_{del}, δ_{deb}, δ_{Cc} are parameters indicating the presence/absence of the different operations of felling and processing (delimbing, debarking, and cross cutting);
- $\tilde{C}_j^{FT}$ is the unit cost for the forest primary transportation phase, [€ h^{-1}], associated to each slope class (i.e., primary transportation technique), j=1,…, J;
- $Pr_{i,j}^{FT}$ is productivity for the primary transportation phase, considering each parcel and slope class;
- $\tilde{C}^{T}$ is the unit cost for the forest transport phase, [€ kg^{-1} km^{-1}];
- d_i^{LW} is the distance from the forest parcel i, $i = 1,\ldots,N$ to the plant location, [km];
- $\tilde{C}_i^{P}$ is the unit cost for the purchasing cost of non forest biomass, [€ kg^{-1}];
- $\tilde{C}_k^{I}$ is the unit installation cost for the k-th technology, [€ MW^{-1} y^{-1}];
- $\tilde{C}_k^{M}$ is the unit maintenance cost for the k-th technology, [€ kg^{-1}];
- ETA_k is the overall plant efficiency for the k-th technology, [%];
- HVP_k is plant capacity, expressed in terms of the maximum developed thermal power for the k-th technology [MW];
- h is the useful working time in a year [hour];
- Ep^{El} is the production of electrical energy;
- Ep^{Th} is the production of thermal energy;
- Ep^{Bd} is the production of bio-oil;
- $\delta_{El,k}$ is a parameter that is set equal to 1 when the plant produces some amount of electrical energy, 0 otherwise;
- $\delta_{Th,k}$ is a parameter that is equal to 1 when the plant produces some amount of thermal energy, 0 otherwise;
- $\delta_{Bd,k}$ is a parameter that is equal to 1 when the plant produces some amount of bio-oil, 0 otherwise;
- ETA_k^{e} is the net electrical efficiency of the k-th conversion technology;
- ETA_k^{t} is the net thermal efficiency of the k-th conversion technology;
- ETA_k^{b} is the net bio-diesel efficiency of the k-th conversion technology;
- Dt is the amount of required thermal energy [MWh];
- P^{Th}, P^{El}, and P^{Bd} are prices for thermal and electric energy, and bio-oil, respectively [€/MWh], including the renewable energy certificates.

2.3 The constraints

Three different classes of constraints are included in the optimization problem: the restrictions over the forest biomass collection in order to prevent species extinction, the continuity equation at the conversion plant, and the constraint related to plant kind. Specifically,

$$u_{i,j} \le \alpha_i \cdot X_{i,j}^{0} \;\; i = 1,\ldots,N \,, \quad j = 1,\ldots,J \,, \tag{8}$$

where $X_{i,j}^{0}$ [m^3] is the initial biomass present in each parcel i of acclivity j, and α_i [%] is an upper bound for biomass collection,

$$HVP_k = \sum_i \sum_j LHV_i \left(u_{i,j} VM_i \frac{1}{3600\, h} \right), \qquad k = 1, \ldots, K, \tag{9}$$

where 3600 is the number of seconds in one hour, *VM* is volumetric mass, and *LHV* is low heating value, and

$$\sum_{k=1}^{K} \delta_k = 1 \ . \tag{10}$$

3. PSO AND PROPOSED DERIVATIVE

A swarm consists of a set of particles moving within the search space, which is *D*-dimensional, each representing a potential solution of the problem. Each particle has a position vector, $X_i = (x_{i1}, \ldots, x_{iD})$, a velocity vector, $V_i = (v_{i1}, \ldots, v_{iD})$ and the position at which the best fitness was encountered by the particle, $Y_i = (y_{i1}, \ldots, y_{iD})$. In each cycle of the evolution the position of the best particle, Y^* is identified. Let *N* be the number of particles in the swarm.

3.1 Manipulation of particles

In each generation, the velocity of each particle is updated by means of its velocity history, its best encountered position and the best position encountered by any particle:

$$V_i = \omega V_i + c_1\, rand(\,)\, (Y_i - X_i) + c_2\, rand(\,)\, (Y^* - X_i), \tag{11}$$

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

$$V_{\min} \le V_j \le V_{\max}, \tag{12}$$

to control excessive roaming of particles outside the search space. Usually $V_{\min}$ is taken as $-V_{\max}$.

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

$$X_i = X_i + V_i. \tag{13}$$

The parameters are as follows: ω is a factor of inertia suggested by Shi and Eberhart [1998] that controls the impact of the velocity history into the new velocity. Acceleration parameters c_1 and c_2 are typically two positive constants, called cognitive and social parameter, respectively. *rand*( ) represents a function which creates random numbers between 0 and 1; two independent random numbers enter Equation (11), which are used to maintain the diversity of the population.

3.2 Manipulation of parameters

The role of the inertia, ω, in (11), is considered critical for the PSO algorithm’s convergence behavior. Although initially the inertia was constant it may vary from one cycle to the next. As it permits to balance out global and local searches, it was suggested to have it decrease linearly with time, usually in a way to first emphasize global search and then, with each cycle of the iteration, prioritize local search, [Shi and Eberhart, 1999]. A significant improvement in the performance of PSO with the decreasing inertia weight over the generations is achieved by using [Jin et al., 2007]

$$\omega = 0.5 + \frac{1}{2(\ln(k)+1)}, \tag{14}$$

where k is the iteration number. In the framework herein described this parameter is adaptively controlled by using (14).

However, the acceleration coefficients and the clamping velocity are neither set to a constant value, like in standard PSO, nor set as a time varying function, like in adaptive PSO variants [Ratnaweera and Halgamuge, 2004; Aramugan and Rao, 2008]. Instead they are incorporated to the own optimization problem. Each particle will be allowed to self-adaptively set its own parameters by using the same process used by PSO given by equations (11) and (13). To this end, these three parameters are considered as three new variables that are incorporated to 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:

$$X_i = (x_{i1}, \ldots, x_{iD}, x_{iD+1}, \ldots, x_{iD+P}). \quad (15)$$

It is clear that the first D variables correspond to the real position vector of the particle in the search space, while the last P account for its personal acceleration constants and velocity limit.

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

Also, V_i and Y_i, giving the velocity and best so far position for particle i, increase their dimension, with corresponding meaning:

$$V_i = (v_{i1}, \ldots, v_{iD}, v_{iD+1}, \ldots, v_{iD+P}), \quad (16)$$

and

$$Y_i = (y_{i1}, \ldots, y_{iD}, y_{iD+1}, \ldots, y_{iD+P}). \quad (17)$$

This way, by using equations (11) and (13), each particle will be endowed additionally with the ability to adjust its parameters by aiming to both the parameters it had when it got its best position in the past and the parameters of the leader, which managed to bring this best particle to its privileged position. As a consequence, particles not only use their cognition of individual thinking and the social cooperation to improve their positions but also to improve the way they do it by accommodating themselves to the best known conditions, namely, their conditions when getting the best so far position and the leader's conditions.

Before providing a schematic representation of the proposed algorithm two more observations have to be made.

For one thing, the discussion so far considers the standard PSO algorithm, which is applicable to continuous systems and cannot be used for mixed discrete-continuous problems, like the one we consider here. To tackle discrete variables this algorithm takes integer parts of the flying velocity vector discrete components into account; hence the new discrete velocities V_i are integer and consequently the new position vector components will also be discrete (since the initial position vectors were generated with discrete values for discrete variables). According to this idea, instead of Eq. (11), velocity updating for discrete variables turns out to be:

$$V_i = fix(\omega V_i + c_1\, rand(\,)\, (Y_i - X_i) + c_2\, rand(\,)\, (Y^* - X_i)), \quad (18)$$

where $fix(\cdot)$ implies that we only take the integer part of the result.

For another, in [Montalvo et al., 2008b], PSO was endowed with a re-generation-on-collision formulation, later generalized in [Izquierdo et al., 2007], which further improves the performance of standard discrete PSO. The random regeneration of the many birds that tended to collide with the best birds was shown to avoid premature convergence, as it prevented clone populations from dominating the search. The inclusion of this procedure into the discrete PSO produces greatly increased diversity, improved convergence characteristics and higher quality of the final solutions. The modified algorithm can be given by the following pseudo-code, with k as iteration number.

- $k = 0$
- Generate a random population of M particles: $\{X_i(k)\}_{i=1}^{M}$, according to (15)
- Evaluate the fitness of the particles (only the first D variables enter the fitness function)
- Record the local best locations $\{Y_i(k)\}_{i=1}^{M}$; according to (17) the values of the corresponding parameters are also recorded
- Record the global best location, $Y^*(k)$, and the list of the m best particles to check collisions (including their corresponding parameters)
- While (not termination-condition) do
  - Determine the inertia parameter $\omega(k)$, according to (14)
  - Begin cycle from 1 to number of particles M
    - Start
      - Calculate new velocity, $V_i(k+1)$, for particle i according to (11), and take its integer part (for discrete optimization) for the first D variables, according to (18)
      - Update position, $X_i(k+1)$, of particle i according to (13)
      - Calculate fitness function for particle i and update Y_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
  - $k = k + 1$
- Show the solution given by the best particle

In this study, a population size of M = 100 particles has been used. Also, among the different termination conditions that may be stated, a condition stopping the process if there is no improvement after a pre-fixed number of iterations has been considered.

The performance of the approach here introduced can be observed from the results reported in the next section.

4. PROBLEM SOLUTION

The main decision variables are the $u_{i,j}$, representing the annual biomass quantity harvested in the *j-th* slope class, $j = 1,...,J$, of the *i-th* parcel, $i = 1,...,N$, [m^{3}y^{-1}], which are continuous decision variables. For understanding the problem dimensionality it should be noticed that 506 forest parcels have been considered and each one is divided in 5 sub-parcels having different slopes. It means that there are 506×5 = 2530 decision variables, considering only $u_{i,j}$.

The presence/absence of a technology in a known location is also a decision variable. PSO algorithm did not consider a δ_k variable for every technology indicating its presence or not; instead, PSO considered a discrete decision variable δ, that takes values between 1 and the amount of technologies (5 for this problem), indicating which technology will be used. δ_k would be equal to 1 if and only if $k = \delta$. The cardinality of variable δ multiplies the solution space by the number of technologies to be taken into account. Only for the sake of illustration, if the range of every $u_{i,j}$ should to be discretized into 50 values, the amount of possible solution would be 5×50^{2530}; this number is really huge and gives an idea of the size of the solution space where PSO will be searching for an "optimal solution".

Maximum velocities were established considering variable types:

- Maximum velocity for continuous variables = 20% of variable range
- Maximum velocity for discrete variables = 50% of variable range

In any case, minimum velocity was established as:

- Minimum velocity = - Maximum velocity

The termination condition stopped the process if after 800 iterations no improvement in the solution had been obtained.

As a result, it was decided the most convenient technology to be used in the plant, which turned to be #5, corresponding to FBP (Fast Pyrolysis), and how much biomass should be harvested annually in every ***j***-th slope of every ***i***-th parcel (data not included in this paper). Regarding the used data for calculating, it was decided to use a plant capable of assimilating approximately 67% of total available biomass.

Taken into account the wide degree of generality used to tackle the problem, the use of the considered approach to deal with a similar problem for other regions would be straightforward, provided that the data are available in databases with the same characteristics as the ones used here. Also, adding new terms to the fitness function would represent neither a conceptual problem nor a long or difficult task in terms of programming.

5. CONCLUSION

In this paper, a PSO algorithm has been used to tackle the strategic planning of a biomass supply chain. In particular, the decision problem was to define the optimal amount of biomass to be harvested and the best technology to produce energy from forest biomass in a specific mountain community.

A simple example has been examined using just one plant, however, a software solution were made for more than one plant to be analysed in the same region.

Using the proposed algorithm, it can be considered any kind of relations between variables for evaluating the objective function. Graphical facilities should be useful to be incorporated in the near future for making a better visualization and data input-output.

ACKNOWLEDGEMENTS

This work has been performed under the support of the projects Investigación Interdisciplinar nº 5706 (UPV) and DPI2004-04430 of the Dirección General de Investigación del Ministerio de Educación y Ciencia (Spain), including the support for I+D+i projects from the Consellería de Empresa, Universidad y Ciencia of the Generalitat Valenciana, and FEDER funds. Thanks also to the support of the Beca MAEC-AECI 0000202066 awarded to one of the authors by the Ministerio de Asuntos Exteriores y Cooperación of Spain.

REFERENCES

Arumugam, M.S., and Rao, M.V.C., 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, Applied Soft Computing 8 (2008) 324–336.

Bonabeau, E., Dorigo, M., and Théraulaz, G., From Natural to Artificial Swarm Intelligence, Oxford University Press, New York, 1999.

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

Freppaz, D., Minciardi, R., Robba, M., Rovatti, M., Sacile, R., Taramasso, A. Optimizing forest biomass exploitation for energy supply at regional level. *Biomass and Bioenergy*, 15-25, 2004

Frombo, F., Minciardi, R., Robba M., Rosso, F., Sacile, R. Strategic Optimization of biomass flows for energy production: a logistic point of view. Proceedings of the 5th Biennial International Workshop Advanced in Energy Studies "Perspectives on Energy Future". Porto Venere (SP) Italy, September 2006. http://www.chim.unisi.it/portovenere/

Frombo, F., Minciardi, R., Robba, M., Rosso, F., Sacile, R. "Supply chain optimization of biofuels". Proceedings of the 15th European Biomass Conference and Exhibition" Berlin, Germany, 2007.

Izquierdo, J., Montalvo, I., Herrera, M., and Pérez, R., "A derivative of Particle Swarm Optimization with enriched diversity," submitted to Computational Optimization and Applications, 2007.

Izquierdo, J., Montalvo, I., Pérez, R., Fuertes, V.S. Design optimization of wastewater collection networks by PSO. *Computer & Mathematics with Applications*. doi:10.1016/j.camwa.2008.02.007, 2008.

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

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

Kennedy, J. and Eberhart, R.C., Swarm Intelligence, Morgan Kaufmann, 2001.

Liao, C.J., Tseng, C.T., Luarn, P. A discrete version of particle swarm optimization for flowshop scheduling problems. *Computers and Operations Research*, 34(10), 3099-3111, 2007.

Montalvo, I., Izquierdo, J., Pérez, R., Tung, M.M. Particle Swarm Optimization applied to the design of water supply systems. *Computer & Mathematics with Applications*, doi:10.1016/j.camwa.2008.02.006, 2008a.

Montalvo, I., Izquierdo, J., Pérez, R., Iglesias, P.L. A diversity-enriched variant of discrete PSO applied to the design of Water Distribution Networks. *Engineering Optimization*. DOI:10.1080/03052150802010607, 2008b.

Montalvo, I., Izquierdo, J., Pérez, R., Mora, D., Improved performance of PSO with self-adaptive parameters for computing optimal design of Water Supply Systems, submitted to *Engineering Applications of Artificial Intelligence*, 2008.

Pan, Q.K., Tasgetiren, F., Liang, Y.C. A discrete particle swarm optimization algorithm for the no-wait flowshop scheduling problem. *Computers and Operations Research*, doi: 10.1016/j.cor.2006.12.030, 2007.

Ratnaweera, A., and Halgamuge, S.K., Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficient, IEEE Trans. Evol. Comput. vol. 8 (June (3)) (2004) 240–255.

Shi, Y. and Eberhart, R.C., "A modified particle swarm optimizer," in Proc. of the IEEE Congress on Evolutionary Computation, Piscataway, NJ, 1998, pp. 69-73.

Shi, Y. and Eberhart, R.C., Empirical study of particle swarm optimization, Proceedings of Congress on Evolutionary Computation, IEEE Service Center, Piscataway, NJ, 1999.