Jul 1st, 12:00 AM

A Particle Swarm Optimization derivative applied to cluster analysis

José Luis Díaz

M. Herrera

Joaquín Izquierdo

Idel Montalvo

R. Pérez

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

A Particle Swarm Optimization derivative applied to cluster analysis

J.L. Díaz, M. Herrera, J. Izquierdo, I. Montalvo, R. Pérez

Centro Multidisciplinar de Modelación de Fluidos, Universidad Politécnica de Valencia, Camino de Vera s/n, 46022, Valencia, Spain
(jldiaz,mahefe,jizquier,imontalvo,rperez@gmmf.upv.es)

Abstract: Modern machine learning and data analysis hinge on sophisticated search techniques. In general, exploration in high-dimensional and multi-modal spaces is needed. Some algorithms that imitate certain natural principles, the so-called evolutionary algorithms, have been used in different aspects of Environmental Science and have found numerous applications in Environmental related problems. In this paper we apply a derivative of PSO (Particle Swarm Optimization), recently introduced by the authors to partitional clustering of a real-world data set obtained from a Water Supply Company. The PSO derivative we consider here improves 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. 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. Finally, the cumbersome aspect, common to all metaheuristics, of choosing the right parameter values is tackled through self-adaptive dynamic parameter control.

Keywords: Partitional clustering; Optimization; Evolutionary algorithm; Particle Swarm Optimization.

1. INTRODUCTION

Modern machine learning and data analysis hinge on sophisticated search techniques. Computer systems that are able to extract information from large amounts of data, that is to say, to perform Data Mining tasks, like pattern recognition, classification, diagnosis, etc. and, in general, systems that are adaptive and show capacity to learn, fundamentally rely on effective and efficient search techniques. Any adaptive system needs some kind of search mechanism in order to explore a feature space describing all possible states of the system. Due to the characteristics of many feature spaces exploration in high-dimensional and multi-modal spaces is needed.

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. Some algorithms that imitate certain natural principles, evolutionary algorithms like Genetic Algorithms, Ant Colony Optimization, Particle Swarm Optimization, Harmony Search, etc., have been used in different aspects of Environmental Science and have found numerous applications in Environmental related problems [Downing, 1998; López, 2001; Nishida et al., 2004; Vojinovic and Solomatine, 2005; Afshar and Mariño, 2006; Crowe et al., 2006; Katsifarakis and Petala, 2006; Valdés

and Barton, 2007; Karterakisa et al., 2007; Montalvo et al., 2008a, 2008b; Izquierdo et al., 2007, 2008a, 2008b].

One of the evolutionary algorithms that has shown great potential and good perspective for the solution of various optimization problems [Dong et al., 2005; Janson et al., 2008; Jin et al., 2007; Liao et al., 2007; Pan et al., 2007; Montalvo et al., 2008a, 2008b; Izquierdo et al., 2008b] is Particle Swarm Optimization (PSO). 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 behaviour and emergent intelligence that arise in socially organized populations.

In this paper we apply a derivative of PSO, recently introduced by the authors [Izquierdo et al., 2008b; Montalvo et al., 2008a, 2008b] to partitional clustering of a real-world data set obtained from a Water Supply Company. In addition, we endow this PSO derivative with a self-adaptive feature that manages to internally control its parameters.

Clustering analysis [Everitt, 1980] plays an important role in many fields and can be used both for preliminary and descriptive data analysis and unsupervised classification [Hastie et al., 2001], and to summarize common features of groups of elements, like identification of centroids or baricenters. Central to all of the goals of cluster analysis is the notion of *similarity*, in terms of proximity, between the individual objects being clustered (otherwise, *dissimilarity* is used to explain the difference). A clustering method attempts to group the objects based on the definition of similarity supplied to it. In the present paper we will work with clusters based in a mixed dissimilarity. For this reason we will use medoids like a representative grouping element (understanding medoid as the object of a cluster, whose average dissimilarity to all the objects in the cluster is minimal).

The database utilized consists in a record of requests, complains and claims (PQR's in Spanish), for the year 2006 of Calarcá Water Supply Company Multipropósito, S.A. ESP, located in a town of the coffee Colombian region. These records are combined with the information of the network model for this year. The results obtained in this work are important to aid the management of the network and the decision making of most relevant places to be considered in future rehabilitations.

The PSO derivative we consider here is adapted to consider mixed discrete-continuous optimization since the problem we tackle here involves the use of both continuous and discrete variables and will work jointly with statistical clustering criteria arranged to take these type of mixed measures. Also, 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. Finally, the cumbersome aspect, common to all metaheuristics, of choosing the right parameter values is tackled through self-adaptive dynamic parameter control.

The remainder of this paper is organized as follows. Next section presents 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. Then, the database under consideration is concisely described and the main objectives in this study presented. Next, section 4 introduces the necessary statistical clustering criteria including the description of the fitness evaluation and the search space. Finally, the main results are reported. A conclusions section wraps up the paper.

2. PSO AND THE 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 of the N particles in the swarm, Y^* is identified.

2.1 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:

$$V_i = \omega V_i + c_1\, rand(\,)\,(Y_i - X_i) + c_2\, rand(\,)\,(Y^* - X_i), \quad (1)$$

On each dimension, particle velocities are clamped to minimum and maximum velocities to control excessive roaming of particles outside the search space:

$$V_{\min} \le V_j \le V_{\max}. \quad (2)$$

Usually $V_{\min}$ is taken as $-V_{\max}$, which is a user defined parameter.

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

$$X_i = X_i + V_i. \quad (3)$$

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*( ) generates two independent random numbers between 0 and 1, and are used to maintain the diversity of the population.

2.2 Manipulation of parameters

The role of the inertia, ω, in (1), is considered critical for the PSO algorithm's convergence behaviour. 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]

$$\omega = 0.5 + \frac{1}{2(\ln(k)+1)}, \quad (4)$$

where k is the iteration number. In the framework herein described this parameter is adaptively controlled by using (4).

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 (1) and (3). 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:

$$X_i = (x_{i1}, \ldots, x_{iD}, x_{iD+1}, \ldots, x_{iD+P}). \quad (5)$$

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 parameters. 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:

$$V_i = (v_{i1}, \ldots, v_{iD}, v_{iD+1}, \ldots, v_{iD+P}) \text{ and} \quad (6)$$

$$Y_i = (y_{i1}, \ldots, y_{iD}, y_{iD+1}, \ldots, y_{iD+P}). \quad (7)$$

This way, by using equations (1) and (3), each particle will be endowed additionally with the ability of adjusting 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 take 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 component velocities V_i are integer and consequently the new position vector discrete components will also be integer (since the initial position vectors were generated with integer values). According to this idea, velocity updating for discrete variables turns out to be:

$$V_i = fix(\omega V_i + c_1\, rand(\,)\,(Y_i - X_i) + c_2\, rand(\,)\,(Y^* - X_i)), \tag{8}$$

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

For another, in [Montalvo et al., 2007b], PSO was endowed with a re-generation-on-collision formulation, 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.

---

1) $k = 0$
2) Generate a random population of M particles: $\{X_i(k)\}_{i=1}^{M}$, according to (5)
3) Evaluate the fitness of the particles (only the first D variables enter the fitness function)
4) Record the local best locations $\{Y_i(k)\}_{i=1}^{M}$; according to (7) the values of the corresponding parameters are also recorded
5) Record the global best location, $Y^*(k)$, and the list of the m best particles to check collisions (including their corresponding parameters)
6) While (not termination-condition) do
   a) Determine the inertia parameter $\omega(k)$, according to (4)
   b) Begin cycle from 1 to number of particles M
      Start
         (1) Calculate new velocity, $V_i(k+1)$, for particle i according to (1), and take its integer part (for discrete optimization) for the first D variables, according to (8)
         (2) Update position, $X_i(k+1)$, of particle i according to (3)
         (3) Calculate fitness function for particle i and update Y_i
         (4) 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
         (5) If particle i is not currently the best particle but coincides with the best, then re-generate particle i randomly (including its parameters)
      End
   c) $k = k + 1$
7) 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.

3. THE DATABASE

The database utilized consists in a record of requests, complains and claims (PQRs in Spanish) issued in Calarcá Water Supply Company Multipropósito, S.A. ESP, for the year 2006. The municipality of Calarcá (Colombia) is located in the Andean area, it has a land area of 21,923 ha; 244 ha belongs to urban zones and 21,679 ha to rural sectors. Population is about 73,500 inhabitants.

These PQRs are reports of the users both in principal and domiciliary network sections. The PQRs indicated the type or description of damages, their locations, relevant technical concepts and the solutions. There were 846 records registered in that year.

Every record was located in a chart of the water network according to the address referenced in the PQR record; addresses were only in terms of street names and numbers, so a hard work had to be done for obtaining the UTM coordinates of every problem reported.

The main pre-processing task involved the selection of relevant and non relevant fields in the database. Breakage dates and times were excluded since their occurrence was deemed to depend strongly on the physical and working conditions of the network. Also, the names of personnel on duty, the repair date and the theoretical roughness of the pipe were neglected. Decision about what fields to include were made based on hydraulic criteria. Since rehabilitation was the main objective, geographical locations suggesting causes and occurrence of water loses were assessed of paramount importance. As a consequence, the information used included: pipe identification, to assess if it was subjected to a high or low number of faults; upstream and downstream node identification, to evaluate concurrence of faulty pipes on the same node pointing to pressure or demand problems at the node; type of reported breakage, either domiciliary or on the main network; pipe diameter; pipe length; pipe material and magnitude of the leak. In addition, data obtained from the mathematical model of the network were included in the database. Specifically, information related with demand characteristics and patterns. However, pressure data were not included, since pressure, being a decisive agent of water loses and breakages, would have blurred all the other specifically sought causes of the problem under consideration, more connected with materials, lengths, diameters, demand patterns, etc., in close relationship with rehabilitation purposes. Finally, also the UTM coordinates of the fault points were included.

As a matter of fact, typical pre-processing tasks for identifying outliers, missing values, etc., were performed. As a consequence, some records were modified, withdrawn, completed, etc.

4. STATISTICAL MEASURES AND FITNESS EVALUATION

4.1 Introduction

Clustering is the grouping of similar objects [Everitt, 1980]. An object can be described by a set of measurements or by its relation to other objects. The goals of cluster analysis are varied and include wide activities such looking for “natural” groups, hypothesis generation etc. Central to all of the goals of cluster analysis is the notion of similarity, in terms of proximity, between the individual objects being clustered (otherwise, dissimilarity is used to explain the difference). A clustering method attempts to group the objects based on the definition of similarity or dissimilarity supplied to it.

4.2 Dissimilarities

The dissimilarity between two objects measures how different they are [Hastie et al., 2001]. It has to be noted here that, although usual metrics can be used, they must not necessarily verify the triangle inequality.

The computation of the dissimilarity between two objects depends on the type of the original variables. Many data sets contain variables of different types. The next method solves the computation of the dissimilarity in a general form, considering that the data set contains p variables:

$$d(i,j) = \frac{\sum_{f=1}^{p} \partial_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^{p} \partial_{ij}^{(f)}} \quad \in [0,1] \tag{9}$$

With $\partial_{ij}^{(f)} = 0$ if x_{if} or x_{jf} is missing, or if $x_{if} = x_{jf} = 0$ and f is an asymmetric binary variable. Otherwise, $\partial_{ij}^{(f)} = 1$. $d_{ij}^{(f)}$ is the contribution of variable f, which depends on its type:

1. If f is binary or nominal, $d_{ij}^{(f)} = 0$ if $x_{if} = x_{jf}$ and $d_{ij}^{(f)} = 1$ otherwise.
2. If f is interval-scaled, $d_{ij}^{(f)} = \frac{|x_{if} - x_{jf}|}{\max_h(x_{hf}) - \min_h(x_{hf})}$.
3. For ordinal and ratio-scaled variables, ranks r_{if} and $z_{if} = \frac{r_{if} - 1}{M_f - 1}$ are computed, and then z_{if} is treated as interval-scaled.

4.3 Clustering algorithms

The goal of cluster analysis is to partition the observations into groups so that the pair-wise dissimilarities between those assigned to the same cluster tend to be smaller than those in different clusters. Among others, clustering algorithms can be classified in two categories: partitioning and hierarchical algorithms. Most partitional clustering algorithms assume an a priori number of clusters, c, and a partition of the data set into c clusters. To get the correct partition, an objective function must be formulated that measures how good a partition is with respect to the data set. Hierarchical clustering algorithms transform a proximity data set into a tree-like structure. The main drawbacks of these algorithms are its high computational cost and that always suffer from the problem of not knowing where to cut the generated tree.

In real-life problems very large data sets containing variables of several types are typically found. This requires for a clustering algorithm to be scalable and capable of handling different attribute types. Classical methods are not the answer: for example, PAM (Partitioning Around Medoids) algorithm [Kaufman & Rousseeuw, 1990] can handle various attribute types but is not efficient with large data sets. k-means algorithms [Hartigan & Wong, 1979; Likas et al., 2003] can handle large data sets but deal with only data sets formed from interval-scaled variables. CLARA (Clustering Large Applications) algorithm [Kaufman & Rousseeuw, 1990] is a combination of sampling approach and the PAM algorithm. Instead of finding medoids, each of which is the most located object in a cluster for the entire data set, CLARA draws a sample from the data set and uses the PAM algorithm to select an optimal set of medoids from the sample [Wei et al., 2003]. To alleviate sampling bias, CLARA repeats the sampling and clustering process multiple times and selects the best set of medoids as the final clustering. The objective function is the next:

$$Cost(M,D) = \frac{\sum_{i=1}^{n} d(O_i, rep(M, O_i))}{n} \tag{10}$$

where D is the data set to be clustered, n the number of objects in D, M is a set of selected medoids, $d(O_i,O_j)$ is the dissimilarity between objects O_i and O_j, and $rep(M,O_i)$ returns the medoid in M which is closest to O_i.

4.4 Statistical clustering criteria

Several clustering criteria have been proposed and three of them are based on the fundamental matrix equation: $T = W + B$, where W and B are the within-cluster and between-cluster variation, respectively. T is, then, the total scatter matrix. For univariate data the above expression reduces to usual ANOVA decomposition. Under this point of view, the ideal form of T is a matrix built with a small W and a large B, so that the distances within the clusters are small compared with distances between clusters medoids. Then, an intuitive procedure for choosing clusters is to minimize the "size" of W and/or maximize B.

The statistical criteria to measure the adequacy of the partition and define the optimization problem to solve the clustering paradigm via heuristics are based in W and B. Friedman and Rubin [1967] proposed minimizing the W trace. Another idea is to minimize $det(W)$ or maximize $trace(BW^{-1})$. More recently, McGregor et al. [2004] have developed new methodologies for validation results based in W too. Barbará et al. [2002] have worked with entropy based measures for categorical data clustering.

5. RESULTS

The PSO algorithm was run several times and results were almost identical. A population size of only 30 particles was used. Maximum and minimum velocities were established as:

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

The termination condition stopped the process if after 20 iterations no improvement in the solution had been obtained. Results were obtained in a mean value of only 40 iterations.

Clusters were made using PSO and considering different possibilities: 2, 3 and 4 clusters. The search space was multi-dimensional and all dimensions were not in correspondence with the same type of variable. Clusters were analysed and some conclusions can rapidly be drawn:

- Two main different groups were identified; the first one was represented mainly by concrete and PVC pipes, while most of the pipes were made of PVC in the second one.
- Pipes length was established as short, medium and long, based on certain mean values and the range of lengths determined for all the pipes in the database. In the first group, most of the pipes were long, and short pipes were less significative. In the second group the amount of medium and long pipes was almost the same and short pipes were more significative.

In correspondence with cluster analysis, it was shown that problems were concentrated on medium size pipes made of concrete and on large size pipes made of PVC. Also some problems on short pipes made of PVC were detected. Either the rest of materials or medium PVC pipes did not have relevant influence on the problems detected.

6. CONCLUSIONS

Data mining analysis helped to discover where most of the problems in a real water distribution network were concentrated. Cluster analysis was carried out using the PSO algorithm. Results can be used as a strategic plan for network rehabilitation, considering that attention should go first where problems seems to appear more frequently.

Richer results could be obtained incorporating some new fields in the database. In that case, it would be necessary to use a bigger population size for solving the problem, but with no added conceptual difficulty.

PSO algorithm was compared to partitional clustering performed by other algorithms that work with various attribute types (such as PAM and CLARA). There were not significant differences on what was obtained, while PSO efficiency was superior. Thus those results

show the PSO's ability in cluster building. Bigger sizes of the database would enhance PSO superiority in efficiency terms (time of execution and best optimization solution), avoiding the need of adapting to new types of data, as happened with PAM, *k*-means and CLARA. However, PSO clustering should also be compared with CLARANS (CLARA based on Randomized Search) and bagged clustering procedures, based on the bootstrap re-sampling.

Searching for knowledge in databases is really necessary for the water supply sector. Incorporating a tool for clustering analysis to the pool of software packages related to water distribution networks could be very useful. However, it should be also important to incorporate some tools for analysing clustering results.

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 Grant MAEC-AECI 0000202066 awarded to one of the authors by the Ministerio de Asuntos Exteriores y Cooperación of Spain.

A special gratitude must be acknowledged to Calarcá Water Supply Company Multipropósito, S.A. ESP for the collaboration with the data used in this work.

REFERENCES

Afshar, M.H., and Mariño, M.A., Application of an ant algorithm for layout optimization of tree networks, *Engineering Optimization*, 38(3), 353–369, 2006.

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.

Barbará, D.; Couto, J.; Li, Y. (2002) "An entropy-based algorithm for categorical clustering" Proceedings of on Information and Knowledge Management (CIKM, 2002)

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

Crowe, A.M., McClean, C.J., and Cresser, M.S., An application of genetic algorithms to the robust estimation of soil organic and mineral fraction densities. *Environmental Modelling & Software*, 21, 1503-1507, 2006.

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

Downing, K., Using evolutionary computational techniques in environmental modelling. *Environmental Modelling & Software*, 13 (5-6), 519-528, 1998.

Everitt, B. (1980) "Cluster Analysis" *Biometrics*, Vol. 37, No. 2 (Jun., 1981), pp. 417-418

Hartigan, J. A. and Wong, M. A. (1979) "A K-means clustering algorithm" Applied Statistics 28, 100–108

Hastie, T; Tibshirani, R.; Friedman, J. (2001) "Elements of Statistical Learning: Data Mining, Inference and Prediction" Springer-Verlag, New York

Izquierdo, J., López, P.A., Martínez, F.J., Pérez, R. Fault Detection in Water Supply Systems by Hybrid (Theory and Data-Driven) Modelling. Mathematical and Computer Modeling, 46/3-4, pp. 341-350, 2007.

Izquierdo, J., Díaz, J.L., Pérez, R., López, P.A., and Mora, J.J., Knowledge Discovery in Environmental Data. In: Meire, P., Coenen, M., Lombardo, C., Robba, M., Sacile, R. (Eds.), Integrated Water Management, vol. 80, Springer, 2008a.

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, 2008b.

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

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

Karterakisa, S.M., Karatzasa, G.P., Nikolosb, I.K., and Papadopouloua, M.P., Application of linear programming and differential evolutionary optimization methodologies for the solution of coastal subsurface water management problems subject to environmental criteria. *Journal of Hydrology*, 342, 3-4, doi:10.1016/j.jhydrol.2007.05.027

Katsifarakis, K.L., and Petala, Z., Combining genetic algorithms and boundary elements to optimize coastal aquifers' management. *Journal of Hydrology*, 327(1-2), 200-207, 2006.

Kaufman, L. and Rousseeauw, P. J. (1990) "Finding gropus in data: an introduction to cluster analysis" Wiley, New York

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

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

Likas, A.; Vlassis, N.; Vebeek, J. L. (2003) "The global K-means clustering algorithm" Pattern Recognition 36, pp. 451 – 461

López, P.A., Metodología para la calibración de modelos matemáticos de dispersión de contaminantes incluyendo regímenes no permanentes. Doctoral Dissertation. Polytechnic University of Valencia (Spain), 2001.

McGregor, A.; Hall, M.; Lorier, P.; Brunskill, J. (2004) "Flow Clustering Using Machine Learning Techniques" Lecture Notes in Computer Science (Passive and Active Network Measurements), vol. 3015/2004, Springer - Berlin

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.

Nishida, W., Noguchi, M., Matsushita, H., and Solomatine, D.P., A Study on the Application of Genetic Algorithm to Calibration of Water Quality Model. *Ann. J. of Hydraulic Engineering* 48(2), 1321-1326, 2004.

Pan, Q.K., Tasgetiren, F., and 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.

Valdés, J., and Barton, A., Multi-objective Evolutionary Optimization of Neural Networks for Virtual Reality Visual Data Mining: Application to Hydrochemistry. *Proceedings of 2007 IEEE International Joint Conference on Neural Networks*. Orlando, Florida, USA. August 12-17, 2007.

Vojinovic, Z. and Solomatine, D.P., Multi-criterial global evolutionary optimisation approach to rehabilitation of urban drainage systems. *Geophysical Research Abstracts*, Vol. 7, 10720. EGU General Assembly, Vienna, 2005.

Wei, C.; Lee, Y.; Hsu, C. (2003) "Empirical comparison of fast partitioning-based clustering algorithms for large data sets" Experts Systems with Applications 24, pp. 351 – 363