Journal of Hydroinformatics | 13.3 | 2011

Swarm intelligence for groundwater management optimization

A. Sedki and D. Ouazar

ABSTRACT

This paper presents some simulation–optimization models for groundwater resources management. These models couple two of the most successful global optimization techniques inspired by swarm intelligence, namely particle swarm optimization (PSO) and ant colony optimization (ACO), with one of the most commonly used groundwater flow simulation code, MODFLOW. The coupled simulation–optimization models are formulated and applied to three different groundwater management problems: (i) maximization of total pumping problem, (ii) minimization of total pumping to contain contaminated water within a capture zone and (iii) minimization of the pumping cost to satisfy the given demand for multiple management periods. The results of PSO- and ACO-based models are compared with those produced by other methods previously presented in the literature for the three case studies considered. It is found that PSO and ACO are promising methods for solving groundwater management problems, as is their ability to find optimal or near-optimal solutions.

Key words | global optimization technique, groundwater resources management MODFLOW, simulation, simulation–optimization model, swarm intelligence

A. Sedki (corresponding author)
D. Ouazar
Department of Civil Engineering,
Ecole Mohammadia d'Ingènieurs,
Université Mohammed V-Agdal,
765 Agdal, Rabat,
Morocco
E-mail: *asedki@emi.ac.ma*

INTRODUCTION

Due to inherent weaknesses of traditional optimization methods for solving complex groundwater management problems, especially discontinuous, or highly nonlinear and nonconvex problems, interest in developing heuristic search methods has grown rapidly in the past decades. The most widely used algorithms including the genetic algorithm (GA) (McKinney & Lin 1994; Wang & Zheng 1998; Cheng *et al.* 2000) and simulated annealing (SA) (Marryott *et al.* 1993; Rizzo & Dougherty 1996; Wang & Zheng 1998). Recently, particle swarm optimization (PSO) (Kennedy & Eberhart 1995) and ant colony optimization (ACO) (Dorigo *et al.* 1996) have been successfully applied to a wide range of engineering and science problems (Dorigo & Stützle 2004; Clerc 2006). Application of PSO and ACO, however, to water resources problems is of more recent origin. Wegley *et al.* (2000) used PSO to determine pump speeds to minimize the total costs in water distribution systems. Abbaspour *et al.* (2001) employed ACO for estimating the unsaturated soil hydraulic parameters. Maier *et al.* (2003) compared the performance of the ACO algorithm with that of GAs for the optimization of water distribution networks. More recently, Li & Chan Hilton (2007) used ACO for optimizing groundwater monitoring network and Montalvoa *et al.* (2008) applied PSO to optimize water distribution systems designs.

In this study, the potential of PSO and ACO to solve groundwater management problems is explored using three benchmark case studies. The objectives are:

(1) To develop a formulation for applying ACO and PSO for some groundwater management problems.

doi: 10.2166/hydro.2010.163

(2) To evaluate the performance of ACO and PSO in preparation for a real case study.

GROUNDWATER MANAGEMENT MODEL

In a groundwater management model there are two sets of variables: state variables and decision variables. The state variables are hydraulic head, which is the dependent variable in the groundwater flow equation. Decision variables include the well locations and pumping rates. The purpose of the management model is to identify the best combination of these decision variables in order to minimize (or maximize) a management objective with respect to constraints. Constraints can refer to bounds on decision variables and state variables. In a groundwater management model, the state variables are defined as a function of the decision variables by a simulation model.

Simulation model

The three-dimensional equation describing the groundwater flow can be expressed as (Harbaugh *et al.* 2000)

$$\frac{\partial}{\partial x}\left(K_x \frac{\partial h}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y \frac{\partial h}{\partial y}\right) + \frac{\partial}{\partial z}\left(K_z \frac{\partial h}{\partial z}\right) + W = S\frac{\partial h}{\partial t} \qquad (1)$$

where h is the hydraulic head; K_x, K_y and K_z are values of hydraulic conductivity along the x, y and z coordinate axes; W is the flux into or out of the system due to sources or sinks; S is the specific storage; and t is time.

Equation (1), together with the flow and/or head conditions at the boundaries of an aquifer system and initial head condition, constitutes the mathematical model for a groundwater flow system. For this work we are using the US Geological Survey code MODFLOW-2000 (Harbaugh *et al.* 2000). MODFLOW is a widely used and well-supported block-centered finite difference code that simulates saturated groundwater.

Optimization model

The object of the management model is to maximize the total pumping from extraction wells in the aquifer or minimize the total cost of pumping, subject to certain constraints including equalities and inequalities. There may be constraints on decision variables and state variables.

For maximization of the total pumping rates, the optimization model can be formulated as

$$\text{Maximize } J = \sum_{i=1}^{T}\sum_{i=1}^{N} Q_i^t \qquad (2)$$

$$\text{Subject to } h_i^t \ge h_{i,\min}^t \quad i = 1, \text{K}, N; t = 1, \text{K}, T \qquad (3)$$

$$q_{i,\min}^t < q_i^t < q_{i,\max}^t \quad i = 1, \text{K}, N; t = 1, \text{K}, T \qquad (4)$$

where J is the objective function; T is the number of management periods; and N is the number of wells. Equation (3) is the hydraulic head constraint where h_i^t is the hydraulic head in well i at management period t and $h_{i,\min}^t$ is the lower head bound in well i at period t. Equation (4) is the well capacity constraint where q_i^t is the pumping rate of well i at management period t; and $q_{i,\min}^t$ and $q_{i,\max}^t$ are the ranges of allowable pumping rates for well i at management period t.

In the minimum cost problem, the objective function is defined as a sum of the capital cost and the operational cost. The cost for operation is assumed to be a function of both the pumping rate and the total lift to bring water from the wellbore to the surface. The capital cost accounts for drilling and installing all the wells. The objective function is given by

$$\text{Minimize } J(Q) = a_1 \sum_{i=1}^{N}\sum_{t=1}^{T} q_i^t + a_2 \sum_{i=1}^{N}\sum_{t=1}^{T} q_i^t d_i + a_3 \sum_{i=1}^{N}\sum_{t=1}^{T} q_i^t (H_i - h_i^t) \qquad (5)$$

$$\text{Subject to } q_{i,\min}^t < q_i^t < q_{i,\max}^t \quad i = 1, \text{K}, N; t = 1, \text{K}, T \qquad (6)$$

$$h_i^t \ge h_{i,\min}^t \quad i = 1, \text{K}, N; t = 1, \text{K}, T \qquad (7)$$

$$\sum_{i=1}^{N} q_i^t \ge Q^t \quad i = 1, \text{K}, N; t = 1, \text{K}, T \qquad (8)$$

$$h_{i_1}^t - h_{i_2}^t \ge \Delta h_{\min}^t \quad i = 1, \text{K}, N; t = 1, \text{K}, T \qquad (9)$$

$$q_{\text{inject}}^t = A \sum_{i=1}^{N} q_i^t + B \quad i = 1, \text{K}, N; t = 1, \text{K}, T \qquad (10)$$

where Equation (5) is the objective function; a_1 is the cost coefficient for well installation; a_2 is the cost coefficient for drilling; a_3 is the cost coefficient for pumping; h_i^t is the hydraulic head in well i at management period t; d_i is the depth of well i; H_i is the land surface elevation at well i. Equation (8) is the demand constraint, where Q_t is the water demand at management period t. Equation (9) is the minimum head difference constraint, where $\Delta h^t_{\min}$ is the specified bound on head differences and $h^t_{i_1}$ and $h^t_{i_2}$ are heads at locations i_1 and i_2 at the period t. This type of constraint is generally associated with the design of capture zones to contain and remove contaminated groundwater. It is typically imposed to force a gradient in the hydraulic flow field. Equation (10) is the balance constraint. This type of constraint is used to maintain a relationship between the injection rate in well i at period t, q^t_{inject}, and the total pumpage, where A and B are the coefficients defining the balance relationship.

To solve the above problems, the constrained model is converted into an unconstrained one by adding the amount of constraint violations to the objective function as penalties:

$$F = J \pm \sum_{i=1}^{NC} P_i \tag{11}$$

and

$$P_1 = \lambda_1 \max(0, h^t_{i,\min} - h^t_i) \tag{12}$$

$$P_2 = \lambda_2 \max(0, \Delta h^t_{\min} - (h^t_{i1} - h^t_{i2})) \tag{13}$$

$$P_3 = \lambda_3 \max(0, Q^t - \sum q^t_i) \tag{14}$$

where F and J are the penalized and non-penalized objective function values, respectively. The minus sign applies for the maximization problem (Equation (2)) and a plus sign for the minimization problem (Equation (5)); P_i ($i = 1, 2, 3$) are the penalty amounts of constraint violation with respect to the hydraulic head constraint in Equations (3) or (7), the minimum head difference constraint in Equation (9) and the demand constraint in Equation (8); NC is the number of constraint violations ($NC = 3$); and λ_i ($i = 1, 2, 3$) are penalty coefficients. Note that if constraint i is satisfied, the max function takes on a value of zero.

Three types of model formulations are considered in this study: (i) maximization of total pumping problem, (ii) minimization of total pumping to contain contaminated water within a capture zone and (iii) minimization of the pumping cost to satisfy the given demand for multiple management periods. Note that, while the first and second problems have been solved for steady-state conditions, the third example is a transient problem. These are described in detail in the fifth section.

USING PSO TO SOLVE GROUNDWATER MANAGEMENT MODEL

Particle swarm optimization (PSO) is a population-based stochastic optimization technique. It was originally proposed by Kennedy & Eberhart (1995) as a simulation of the social behavior of social organisms such as bird flocking and fish schooling. PSO shares many similarities with evolutionary computation techniques such as genetic algorithms (GA). The system is initialized with a population of random solutions and searches for optima by updating generations. However, unlike GA, PSO has no evolution operators such as crossover and mutation. Instead, PSO relies on the exchange of information between individuals (particles) of the population (swarm). In effect, each particle of the swarm has an adaptable velocity (position change), according to which it moves in the search space. The particles update their velocities and positions based on the previous best position and towards the current best position attained by any other member in its neighborhood.

In PSO-based groundwater simulation–optimization code, for an optimization problem with only a single management period, each particle represents all decision variables (pumping rates). However, for an optimization problem with multiple management periods, the pumping rate of any well can vary from one management period to another. In this case, the pumping rates for all the management periods are joined to form one decision vector of a larger dimension. If the number of total candidate wells is M and the number of the management periods is T, the ith particle of the swarm can be represented by a $D = M \times T$-dimensional vector:

$$X_i = (q^1_{i1}, q^1_{i2}, \mathrm{K}, q^1_{iM}, q^2_{i1}, q^2_{i2}, \mathrm{K}, q^2_{iM}, \mathrm{K}, q^T_{i1}, q^T_{i2}, \mathrm{K}, q^T_{iM}) \tag{15}$$

where the subscript in q_i is the well number and the superscript is the management period number. The range of q_i is $q_{i,\min} < q_i < q_{i,\max}$.

The velocity of this particle can be represented by another D-dimensional vector:

$$V_i = (v^1_{i1}, v^1_{i2}, \mathrm{K}, v^1_{iM}, v^2_{i1}, v^2_{i2}, \mathrm{K}, v^2_{iM}, \mathrm{K}, v^T_{i1}, v^T_{i2}, \mathrm{K}, v^T_{iM}) \tag{16}$$

The best position found by the ith particle so far is denoted as

$$P_i = (p_{i1}^1, p_{i2}^1, \mathrm{K}, p_{iM}^1, p_{i1}^2, p_{i2}^2, \mathrm{K}, p_{iM}^2, \mathrm{K}, p_{i1}^T, p_{i2}^T, \mathrm{K}, p_{iM}^T). \quad (17)$$

Defining g as the index of the best particle in the swarm (i.e. the gth particle is the best) and let the superscripts denote the iteration number, then the swarm is manipulated according to the following two equations (Eberhart & Shi 1998):

$$V_i^{iter+1} = \chi(\omega V_i^{iter} + c_1 r_1^{iter}(P_i^{iter} - X_i^{iter}) + c_2 r_2^{iter}(P_g^{iter} - X_i^{iter})) \quad (18)$$

$$X_i^{iter+1} = X_i^{iter} + V_i^{iter+1} \quad (19)$$

where $i = 1,2,\ldots,N$; N is the size of the swarm. c_1 and c_2 are two positive constants named as learning factors and r_1 and r_2 are random numbers in the range (0,1). ω is called the inertia weight and χ is a constriction factor which is used, as an alternative to ω to limit velocity.

Equation (18) is used to calculate the particle's new velocity according to its previous velocity and the distances of its current position from its own best position and the group's best position. Then, the particle flies toward a new position according to Equation (19). Proper fine-tuning of the parameters c_1 and c_2 in Equation (18) may result in faster convergence of the algorithm, and alleviation of the problem of local minima. The constriction factor χ in Equation (18) is used to control the magnitude of the velocities. It is observed that, if the particle's velocity is allowed to change without bounds, the swarm will never converge to an optimum, since subsequent oscillations of the particle will be larger. To control the changes in velocity, Clerc (1999) introduced the constriction factor into the standard PSO algorithm to ensure the convergence of the search. The role of inertial weight ω in Equation (18) is to control the impact of the previous velocities on the current one. A large inertial weight facilitates global exploration (searching new areas), while a small weight tends to facilitate local exploration. Hence selection of a suitable value for the inertial weight ω usually helps in the reduction of the number of iterations required to locate the optimum solution (Parsopoulos & Vrahatis 2002). The variable ω is updated according to

$$\omega = \frac{maxiter - iter}{maxiter} \quad (20)$$

where *iter* is the current iteration number and *maxiter* is the maximum number of allowable iterations.

The steps of the PSO algorithm are explained in the following.

(1) Generate initial position of particles randomly in the range of [$q_{\min}$, $q_{\max}$] and initial velocity in the range of [($q_{\min}$–$q_{\max}$)/2, ($q_{\max}$–$q_{\min}$)/2].
(2) Evaluate the fitness of each particle according to either Equation (2) or Equation (5). P_i is set as the positions of the current particles, while P_g is set as the best position of the initialized particles.
(3) Reduce the inertia weights ω according to Equation (20).
(4) The positions and velocities of all the particles are updated according to Equations (18) and (19); then a group of new particles are generated.
(5) Evaluate the fitness of each new particle, and the worst particle is replaced by the stored best particle. If the new position of the ith particle is better than P_i, then set P_i equal to the new value. If the best position of all new particles is better than P_g, then P_g is updated.
(6) Check the convergence criterion. If the stopping criterion is met, stop; else repeat steps (3)–(5).

Note that, to evaluate Equations (3), (5), (7) and (9), the values of the hydraulic head in the given cells of the model, h_i^t, must be computed for each trial solution generated by PSO. To do this, PSO was linked externally with the MODFLOW. In effect, PSO begins with a random set of trial solutions for pumping rates. For every trial solution (particle) of random pumping rates, MODFLOW is executed once to update the hydraulic head distribution in response to those pumping rates. The output of the simulation consists of the values of hydraulic heads at all model cells. The values of hydraulic heads for the selected observation points are extracted to evaluate the objective function along with the constraints and violation constraints are evaluated by calculating the modified objective function (Equation (11)). Next, new trial solutions for pumping rates are generated based on Equations (18) and (19) and again MODFLOW is called to update the head distribution and the objective function. This process is continued until an optimal solution is reached based on the objective function and the constraints.

USING ACO TO SOLVE GROUNDWATER MANAGEMENT MODEL

Ant colony optimization (Dorigo *et al.* 1996) is a discrete combinatorial optimization algorithm which is inspired by the ability of an ant colony to find the shortest paths between their nest and a food source. This is accomplished by using pheromone (chemical) trails as a form of indirect communication. When searching for food, ants initially explore the area surrounding their nest in a random manner. As soon as an ant finds a food source, it evaluates the quantity and the quality of the food and carries some of it back to the nest. During the return trip, the ant deposits a pheromone trail on the ground. The quantity of pheromone deposited, which may depend on the quantity and quality of the food, will guide other ants to the food source. Thus a shorter path tends to have a higher pheromone density, making it more likely to be chosen by other ants (Bonabeau *et al.* 2000). This shortest path represents the global optimal solution and all the possible paths represent the feasible region of the problem.

Application of the ACO to a combinatorial optimization problem requires that the problem can be projected on a graph (Dorigo *et al.* 1996).

Consider a graph $G = (D,L,C)$ in which $D = d_1,d_2,\ldots,d_n$ is the set of decision points at which some decisions are to be made, $L = l_{ij}$ is the set of options $j = 1,2,\ldots,J$ available at decision points i and finally $C = c_{ij}$ is the set of local costs associated with options $L = l_{ij}$. The components of sets D and L may be constrained if required. A path on the graph, called a solution (ϕ), is then composed of a selection of an option at each decision point. The minimum cost path on the graph is called the optimal solution (ϕ^*). The cost of a solution is denoted by $f(\phi)$ and the cost of the optimal solution by $f(\phi^*)$ (Dorigo & Di Caro 1999).

In the example depicted in Figure 1, there are two nodes, d_1 and d_2. At d_1, four options are available, denoted by l_{11}, l_{12}, l_{13} and l_{14}, if it is assumed that the graph is traversed from d_1 to d_2, and three options l_{21}, l_{22} and l_{23} are available at d_2. For each of the available options is associated a local cost value c_{ij}.

The basic steps of the ACO algorithm (Dorigo *et al.* 1996; Maier *et al.* 2003) may be defined as follows:

(1) A colony of m ants is chosen and the amount of pheromone trail on all options $L = l_{ij}$ are initialized to some proper value.

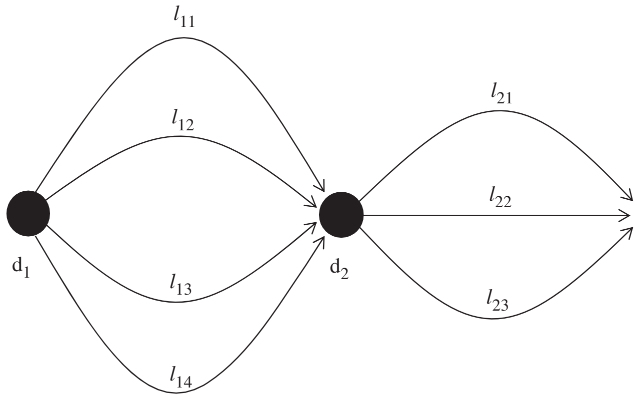

Figure 1 | Typical representation of ant colony optimization graph.

(2) Ant number k is placed on the starting decision point of the problem.

(3) A transition rule is used for ant k currently placed at decision point i to decide which option to select. The transition rule used here is defined as follows (Dorigo *et al.* 1996):

$$p_{ij}(k,t) = \frac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}}{\sum_{j=1}^{J}[\tau_{ij}(t)]^{\alpha}[\eta_{ij}]^{\beta}} \quad (21)$$

where $p_{ij}(k,t)$ is the probability that ant k selects option l_{ij} from the ith decision point at iteration t; $\tau_{ij}(t)$ is the concentration of pheromone on option l_{ij} at iteration t; $\eta_{ij} = 1/c_{ij}$ is the heuristic value representing the local cost of choosing option j at point i; α and β are two parameters which determine the relative influence of the pheromone trail and the heuristic information respectively.

Once the option at the current decision point is selected, ant k moves to the next decision point and a solution is incrementally constructed by ant k as it moves from one decision point to the next one. This procedure is repeated until all decision points of the problem are covered and a complete trial solution ϕ is constructed by ant k.

(4) The cost $f(\phi)$ of the trial solution generated is calculated.

(5) Steps 2–4 are repeated for all ants, leading to the generation of m trial solutions and the calculation of their corresponding costs, referred to as an iteration (t).

(6) After the completion of one iteration (t) (i.e. the construction of m trial solutions), the pheromone trails are

updated by the following rule (Dorigo *et al.* 1996):

$$\tau_{ij}(t+1) = \rho\tau_{ij}(t) + \Delta\tau_{ij} \tag{22}$$

where $\tau_{ij}(t+1)$ is the amount of pheromone trail on option l_{ij} at iteration $t+1$; $\tau_{ij}(t)$ is the concentration of pheromone on option l_{ij} at iteration t; $0 < \rho < 1$ is the coefficient representing the pheromone evaporation and $\Delta\tau_{ij}$ is the pheromone deposit on option l_{ij}. The parameter ρ is used to avoid stagnation of the pheromone trails in which all the ants select the same option at each decision point and it enables the algorithm to "forget" bad decisions previously taken.

Different methods are suggested for calculating $\Delta\tau_{ij}$. The method used here is rank-based ant system (ASrank) (Bullnheimer *et al.* 1999) in which, in each iteration, only the $(w-1)$ best ranked ants and the ant that produced the best solution are allowed to deposit pheromone, i.e.

$$\Delta\tau_{ij} = \sum_{k=1}^{w-1}(w-k)\Delta\tau_{ij}^{k} + w\Delta\tau_{ij}^{\text{best}}. \tag{23}$$

The amount of pheromone change is defined as (Dorigo *et al.* 1996)

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{R}{f(\phi)^{k}} & \text{if option } j \text{ is chosen by ant } k \\ 0 & \text{otherwise} \end{cases} \tag{24}$$

where $f(\phi)^k$ is the cost of the solution produced by ant k and R is a quantity related to the pheromone trail called the pheromone reward factor. Ants deposit an amount of pheromone proportional to the quality of the solutions they produce. Consequently, options that are used by the best ant and which form a part of the lower cost solution, receive more pheromone and therefore are more likely to be chosen by ants in future iterations.

(7) The process defined by steps (2)–(6) is continued until the iteration counter reaches its maximum value defined by the user or some other convergence criterion is met.

Application of the ACO algorithm, as defined previously, to groundwater management problems requires the problem to be defined in terms of a graph G. For this, each well i in each management period t of the problem is considered as the decision point of the graph. The list of available pumping rates for each well in each period constitutes the available options at each decision point of the graph. Here, the options available at each decision point i are originally represented by infinite values of pumping rates in the range defined by $q_{i,\min}$ and $q_{i,\max}$. These options, however, are discretized so that a combinatorial optimization method such as the ACO algorithm can apply. After the discretization, at each decision point there are a number of options corresponding to all possible discrete values resulting from this discretization. For example, for $q_{i,\min} = 0$ and $q_{i,\max} = 7000$ m^3/d and the precision requirement of the solution is 1000 m^3/d, then at each decision point there exist eight options, corresponding to the eight possible discrete values of q_i.

Figure 2 is an example of a simple graph where there are three wells and two management periods. Therefore, there are six decisions points ($d_1^1, d_2^1, d_3^1, d_1^2, d_2^2, d_3^2$), where d_i^t represents the ith managed well in period t. At each decision point d_i^t there are eight options (l_1,l_2,l_3,l_4,l_5,l_6,l_7,l_8) corresponding to the eight possible discrete values of the pumping rate $q_{i,\min}^t \le q_i^t \le q_{i,\max}^t$. Then, for this example, the total number of possible combinations (trial solutions) is 8 × 10^6 and one of the combinations is the optimal pumping solution we look for. The cost of a trial solution is given by either Equation (2) or Equation (5).

The coupling between ACO and MODFLOW is similar to that between PSO and MODFLOW. A colony of m ants is used to traverse the graph sequentially, making decisions at each decision point. At a decision point d_i^t (i.e. well i in period t) an ant selects one of the available option j (i.e. a discrete possible value j of the pumping rate $q_{i,\min}^t \le q_i^t \le q_{i,\max}^t$) based

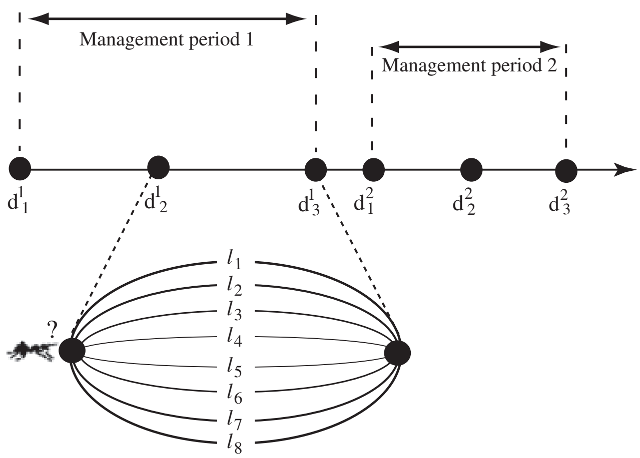

Figure 2 | Representation of groundwater optimization problems in terms of a graph.

on a transition rule given by Equation (21), where the heuristic value η_{ij} is taken here equal to the inverse of the value of pumping rate selected for the well considered in a minimizing problem. However, in a maximizing problem, η_{ij} equal to the pumping rate value and the change in pheromone concentration is $\Delta\tau_{ij}^k = Rf(\phi)^k$.

Initially, each of the m ants have an equal probability to choose a specific option at each decision point (i.e. at initialization, each option has an equal initial pheromone intensity and the parameter β set to zero). Once all ants of the colony have completed the construction of their solution, each one is evaluated using MODFLOW and the modified objective function (Equation (11)) is calculated. After the generation of the m trial solution and the calculation of their corresponding cost, the concentration of the pheromone tails is modified by applying the updating rule Equation (22). The steps of generating trial solutions, calling MODFLOW to evaluate the chosen solutions and updating the pheromone concentrations are repeated until a converged optimal solution is reached.

APPLICATION EXAMPLES

To investigate the performance of applying the PSO and ACO algorithms to solve groundwater management problems, three typical problems are used as examples. They are maximum groundwater supply, hydraulic capture zone design with a single management period, and minimum cost pumping to a multiple management period problem. These example problems were chosen because they have been used as examples for a number of optimization algorithms in previous studies, making it possible to compare our results.

Example 1: maximization of total pumping

The first example is obtained from Example 1 of McKinney & Lin (1994). The management objective is to maximize the yield from a homogeneous, isotropic, unconfined aquifer using 10 pumping wells with constraints on hydraulic head and pumping rates. The hydraulic conductivity and the areal recharge rate are 50 m/d and 0.001 m/d, respectively. The plan view of the discretized aquifer and the potential pumping well locations are shown in Figure 3. The constraints on hydraulic head are that the head must be above zero (bottom) anywhere in the aquifer and the range of the pumping rate is from 0 to 7000 m^3/d for every well. The objective function to be maximized is in the form of Equation (2) with $T = 1$ since only one management period is assumed.

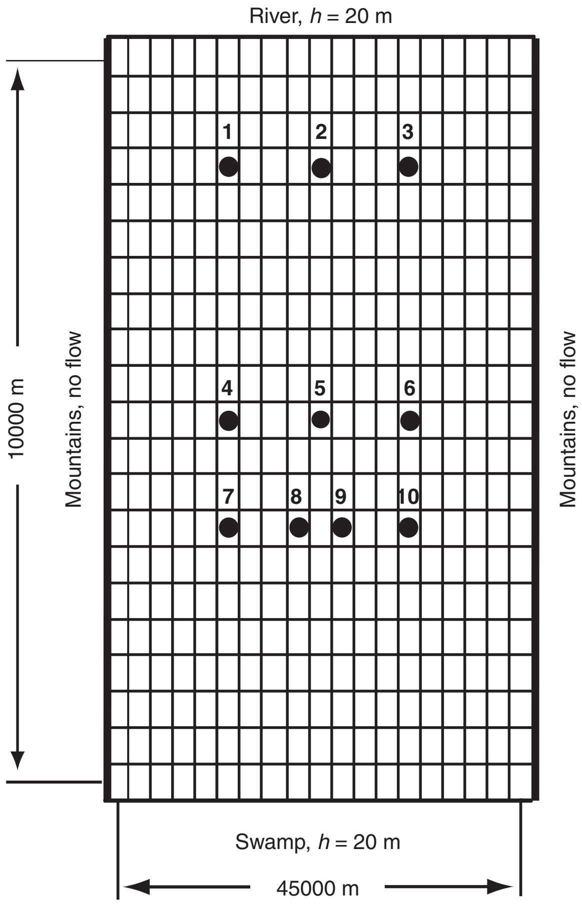

Figure 3 | Aquifer system of Example 1.

McKinney & Lin (1994) solved this problem using linear programming (LP) and genetic algorithms (GA [M&L]). This problem was also solved by Wang & Zheng (1998) using genetic algorithms (GA [W&Z]) and simulated annealing (SA). Wu *et al.* (1999) developed a GA-based SA penalty function approach (GASAPF) to solve this problem. Wu & Zhu (2006) used the shuffled complex evaluation method developed at the University of Arizona (SCE-UA) to solve the same problem. Table 1 compares the results of these studies with those obtained using PSO and ACO algorithms. From Table 1, it is seen that the pumping rates calculated by the PSO and ACO models are in close agreement with the LP solution. Furthermore, the PSO and ACO pumping rates are symmetric due to the symmetry of the aquifer system.

Table 1 | Maximum pumping Example 1: results (units: m^3/d)

Well	LP	GA[M&L]	GA[W&Z]	SA	GASAPF	SCE-UA	PSO	ACO
1	7000	7000	7000	7000	7000	7000	7000	7000
2	7000	7000	7000	7000	7000	7000	7000	7000
3	7000	7000	7000	7000	7000	7000	7000	7000
4	6000	7000	5000	6200	6056	5987	6154	5000
5	4500	2000	5000	4700	4290	4477	4704	5000
6	6000	6000	6000	6200	6056	5986	6146	6000
7	6800	7000	7000	6650	6774	6814	6764	7000
8	4100	4000	4000	4000	4064	4094	3856	4000
9	4100	4000	4000	4000	4064	4094	4175	4000
10	6800	7000	7000	6650	6774	6814	6709	7000
Total pumping	59,300	58,000	59,000	59,400	59,058	59,266	59,508	59,000

The symmetric solutions provide a check on the search accuracy and validity of these algorithms.

In the ACO algorithm, after sensitivity analysis, the model parameters adopted are as follows: population size = 200; $\alpha = 1$, $\beta = -0.1$, $\rho = 0.85$ and $R = 1$. The precision requirement for pumping rate is the same 1000 m^3/d as in McKinney & Lin (1994). As discussed previously, for $q_{min} = 0$ and $q_{max} = 7000$ m^3/d, the decision variables are discretized into scaled integer values ranging from 0 to 7000 in increments of 1000 m^3/d. Figure 4 shows the convergence behavior of both PSO and ACO models. The ACO solution converged to a stable objective function value of 59,000 m^3/d after 28 iterations as shown in Figure 4. The results are also listed in Table 1 from which it can be observed that ACO converged to the same objective function value as GA (Wang & Zheng 1998) and better than that of GA (McKinney & Lin 1994) when the precision requirements for pumping rates are identical.

The PSO algorithm was run with the same population size as in ACO. By trail and error, it was found that the best values for χ, c_1 and c_2 are 0.8, 2 and 2, respectively. From the plot of PSO in Figure 4, it can be observed that the best PSO solution converged to 59,000 m^3/d after only 9 iterations and after 82 iterations, the solution converged to stable maximum of 59,508 m^3/d. The results are also given in Table 1, which showed that the PSO algorithm obtained a better objective function compared to SA, GASAPF and SCE-UA.

In comparison with ACO, as can be seen from Figure 4, PSO presents faster convergence and provides a better solution. Note that the PSO solution has a higher precision than the ACO solution. In PSO, each pumping rate parameter (i.e. continuous parameter) can take any value from the interval defined by $q_{min} = 0$ and $q_{max} = 7000$ m^3/d. However, for the ACO solution, the precision in this example for each parameter is 1000 m^3/d.

Example 2: hydraulic capture zone design

The second example is from case 1 of example 1 of Zheng & Wang (2003). In this example, the objective is to determine

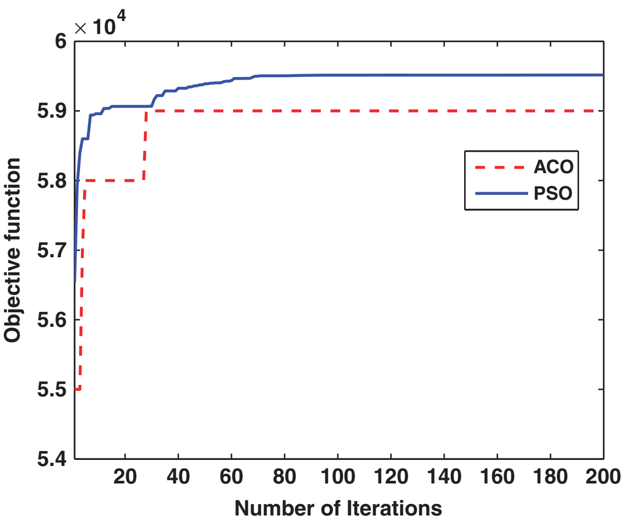

Figure 4 | Comparison of the convergence of PSO and ACO solutions for Example 1.

the minimum amount of pumping required to contain an existing contaminant plume within a capture zone and prevent it from spreading by using wells to control the direction of flow. The aquifer system is assumed to be unconfined with a uniform hydraulic conductivity of 25 m/d. Figure 5 shows the plan view of the aquifer. The finite-difference grid consists of 18 columns and 30 rows with a uniform grid spacing of 100 m in either direction. The filled contours show the location and concentration distribution of the plume and the arrows indicate the location of monitoring points where inward hydraulic gradients are to be maintained. There are four pumping wells, shown as solid dots in Figure 5, which will be used to achieve containment of the plume and two injection wells, shown as triangles in Figure 5, to put back into the aquifer the extracted and treated groundwater. The objective function in Equation (5) is reduced to the first term ($a_1 = 1/\text{m}^4$) and the constraints are Equations (6), (7), (9) and (10). The hydraulic head constraints must be non-negative and all the pumping rates must be in the range of 0 to 5000 m^3/d. The minimum head difference constraint between any two specified model cells (as connected by the arrows in Figure 5) must be greater than or equal to zero. Finally, for the balance constraint as expressed in Equation (10), the injection rates at the two injection wells are each required to be one-half of the total pumpage from the four pumping wells (i.e. $A = -0.5$ and $B = 0$). Zheng & Wang (2003) solved the problem using GA. Table 2 compares the optimal solutions from GA, PSO and ACO. Comparatively, the lower objective function is obtained by the PSO model.

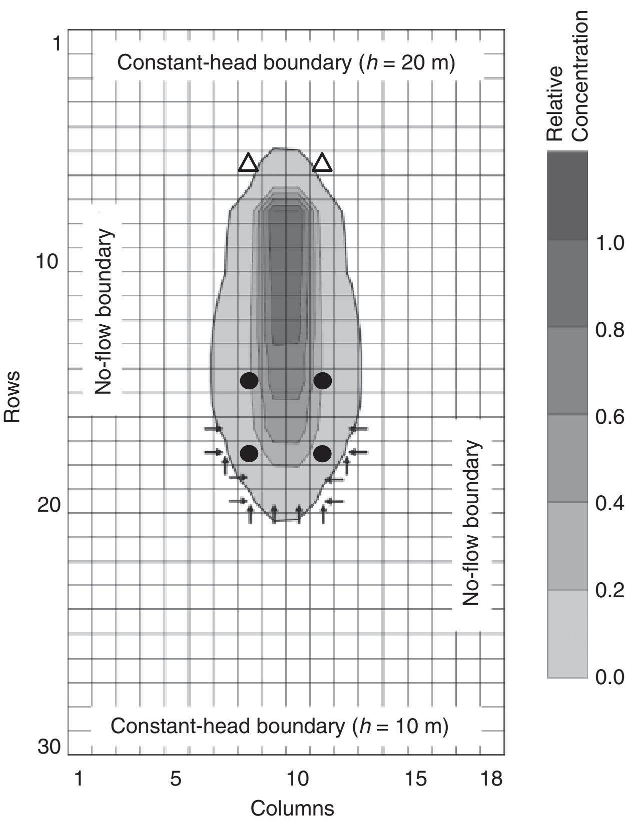

Figure 5 | Aquifer system of Example 2.

Table 2 | Minimum pumping Example 2: results (units: m^3/d)

Well	GA	PSO	ACO
1	161.3	92.00	322.6
2	645.2	94.50	322.6
3	1774.0	1928.0	1774
4	1613.0	1927.0	1774
Total	4193.5	4041.5	4193.2

In PSO, the population size is set to 200 and the other parameters are the same as those for Example 1. In the ACO algorithm, the size of the population is the same as in PSO. After sensitivity analysis, the other parameters were adopted as $\alpha = 1$, $\beta = 0.1$, $\rho = 0.9$ and $R = 1$. Figure 6 shows convergence behavior in the 200 runs of both the PSO and ACO methods. It is observed that PSO outperforms ACO in terms of convergence speed and the quality of the solution. The optimal solutions of both methods are listed in Table 2. It should be noted that, in this example, the precision requirement for the ACO solution is 161 m^3/d.

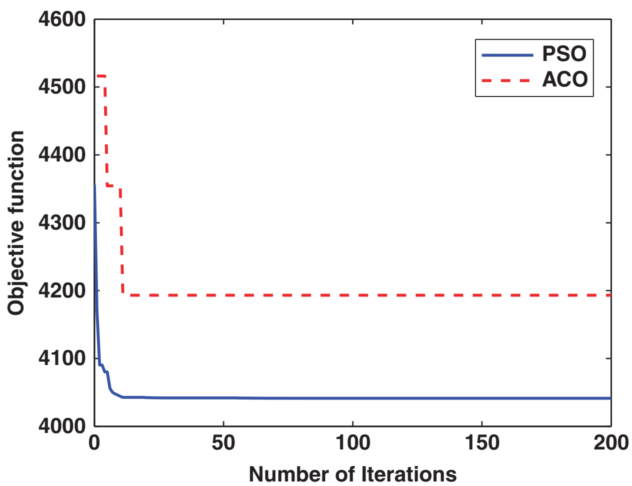

Figure 6 | Comparison of the convergence of PSO and ACO solutions for Example 2.

Compared with GA (Zheng & Wang, 2003), the ACO algorithm found a slightly better objective function value when the precision requirements for pumping rates are identical. The optimal solutions for ACO and GA are shown in Table 2.

Example 3: minimization of the pumping cost for multiple management periods

This example problem is obtained from Jones *et al.* (1987) and Wang & Zheng(1998). The problem is to find optimal pumping rates that would yield the minimum cost from an unconfined aquifer with a hydraulic conductivity of 86.4 m/d and specific yield of 0.1. Figure 7 shows the plan view of the discretized aquifer and the locations of eight potential pumping wells. The model is transient and has an initial head of 100 m everywhere. The distance from the ground surface to the bottom of the aquifer is 150 m at all locations. The objective function in Equation (5) is reduced to the last term only ($a_3 = 1/\text{m}^4$), i.e. the pumping cost, and the constraints are Equations (6)–(8). The number of management periods for this example is 4 and each of them has 91.25 d. The water demands for each management period are 130,000, 145,000, 150,000 and 130,000 m^3/d, respectively. The minimum hydraulic head is zero and the range for each pumping well is from 0 to 30,000 m^3/d. Jones *et al.* (1987) solved the problem using differential dynamic programming (DDP). Wang & Zheng (1998) also solved this problem using genetic algorithms (GA) and simulated annealing (SA). The solution comparisons to the minimum cost pumping example from DDP, GA, SA, PSO and ACO are presented in Table 3.

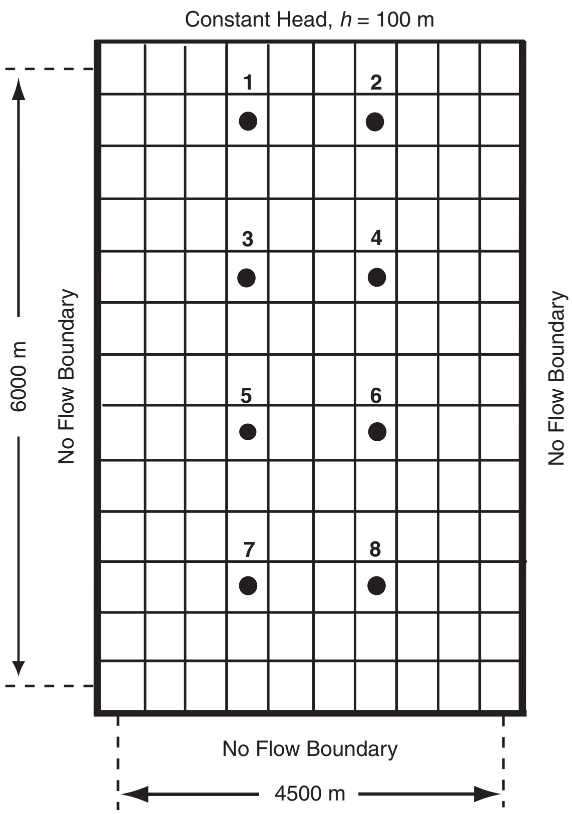

Figure 7 | Aquifer system of Example 3.

As can be seen from Table 3, the total pumping calculated by ACO and PSO is in good agreement with the given water demands for each management period. Furthermore, compared with GA and SA, the best solution was found by ACO followed by PSO. However, DDP gives a better final objective function value than PSO and ACO.

In PSO and ACO, a total of 32 parameters are required, each of which represents the pumping rate at one of the eight potential wells in each of the four management periods. The population size is increased to 600 from 200 in the previous examples to accommodate the increase in the number of parameters. As a consequence, more computational time would be needed.

Using the value of 200 for the maximum number of iterations, both PSO and ACO were run for different trials. In ACO the best parameters found are $\alpha = 1$, $\beta = 0.1$, $\rho = 0.9$ and $R = 1$. For the PSO model, the parameters adopted are $\chi = 0.5$, $c_1 = 1.3$ and $c_2 = 2.7$. Figure 8 shows the convergence behavior of both models, from which it can be observed that the objective function decreases rapidly at the early iterations for both methods with a faster convergence in the case of PSO. However, ACO converged to a lower objective function value than PSO. Table 3 lists the optimal pumping rates from both methods.

DISCUSSION AND CONCLUSIONS

In this paper, the Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) algorithms have been formulated and explored through three example problems to solve

Table 3 | Minimum pumping with multiple management periods Example 3: results (units: m^3/d)

Well	Planning period	DDP	GA	SA	PSO	ACO
1	1	30,000	28,000	30,000	30,000	30,000
2	1	30,000	28,000	30,000	28,996	30,000
3	1	21,924	28,000	17,000	16,827	20,000
4	1	21,924	4000	17,000	16,896	12,000
5	1	7494	12,000	16,000	2049	6000
6	1	7494	12,000	11,000	7647	14,000
7	1	5582	14,000	7000	9767	9000
8	1	5582	4000	2000	17,818	9000
Demand		130,000	130,000	130,000	130,000	130,000
1	2		28,000	30,000	30,000	30,000
2	2		28,000	30,000	29,913	30,000
3	2		10,000	21,000	9866	170,000
4	2		20,000	21,000	23,580	27,000
5	2		8000	10,000	7389	12,000
6	2		14,000	12,000	14,860	12,000
7	2		16,000	12,000	13,838	1000
8	2		20,000	9000	15,554	16,000
Demand			144,000	145,000	145,000	145,000
1	3		28,000	30,000	29,984	30,000
2	3		30,000	30,000	29,992	30,000
3	3		12,000	25,000	19,381	30,000
4	3		28,000	18,000	9756	18,000
5	3		6000	12,000	23,131	10,000
6	3		8000	15,000	16,662	20,000
7	3		26,000	10,000	1801	8000
8	3		12,000	10,000	19,293	4000
Demand			150,000	150,000	150,000	150,000
1	4		28,000	30,000	29,947	30,000
2	4		28,000	30,000	30,000	30,000
3	4		20,000	22,000	22,929	22,000
4	4		22,000	12,000	3174	6000
5	4		4000	5000	12,324	1000
6	4		8000	13,000	9621	14,000
7	4		6000	6000	8162	9000
8	4		14,000	12,000	13,843	18,000
Demand			130,000	130,000	130,000	130,000
Total cost		28,693,336	29,779,432	29,572,110	29,552,000	29,497,050

groundwater management models. In the proposed management model, MODFLOW was used as the simulation component in the coupled simulation–optimization model. The obtained results show that the PSO and ACO models yield identical or better quality solutions when compared to other methods in the literature for the case studies considered.

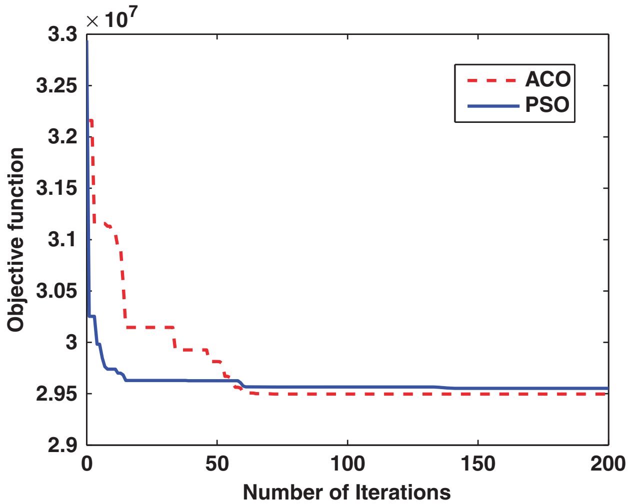

Figure 8 | Comparison of the convergence of PSO and ACO solutions for Example 3.

For the maximization of total pumping problem, results of the PSO are in close agreement with the LP solution and better than the ACO, SA, GASAPF,SCE-UA abd GA. However, ACO obtained the same objective function value as GA (Wang & Zheng 1998) and better than that of GA (McKinney & Lin 1994) when the precision requirements for pumping rates are identical.

For the hydraulic capture problem, the result from the PSO is better than ACO and GA. whereas ACO gives a slightly better objective function value than GA.

To further explore the potential of ACO and PSO models, they are tested for the multiple management period problem. It is found that ACO and PSO models give better quality solutions than those obtained from GA and SA. However, their results are worse than DDP in terms of the final objective function value.

When comparing PSO and ACO, the obtained results show that the convergence is faster in the case of PSO. In addition, for problems with small numbers of optimization parameters, such as Examples 1 and 2 in this study, the PSO algorithm finds better quality solutions than the ACO. However, for problems with large numbers of optimization parameters, such as Example 3, ACO provided the best solution. The better performance of ACO with this larger case study could be attributed to its greater ability to explore, while still exploiting the best information. Also, in ACO the size of the search space is reduced after the discretization.

The results of this study thus demonstrate that an incremental improvement in the groundwater management model can be achieved through the use of swarm-intelligence-based models. The results, based on only using three benchmark case studies from the literature, are extremely promising, but a wider test of ACO and PSO algorithms on more groundwater management problems is required to determine their utility for real case studies.

REFERENCES

Abbaspour, K.C., Schlin, R. & Van Genuchten, M.T. 2001 Estimating unsaturated soil hydraulic parameters using ant colony optimisation. *Adv Wat. Res.* **24**(8), 827–933.

Bonabeau, E., Dorigo, M. & Theraulaz, G. 2000 Inspiration for optimization from social insect behavior. *Nature* **406**, 39–42.

Bullnheimer, B., Hartl, R.F. & Strauss, C. 1999 A new rank based version of the Ant System: a computational study. *Central Europ. J. Oper. Res. Econ.* **7**(1), 25–38.

Cheng, A.H.-D., Halhal, D., Naji, A. & Ouazar, D. 2000 Pumping optimization in saltwater-intruded coastal aquifers. *Wat. Res. Res.* **36**, 2155–2165.

Clerc, M. 1999 The swarm and the queen: towards a deterministic and adaptive particle swarm optimization. In: *Proc. Congress of Evolutionary Computing* vol. 3 (Angeline, P.J., Michalewicz, Z., Schoenauer, M., Yao, X. & Zalzala, A. (Eds.)). IEEE, New York, 1951–1957.

Clerc, M. 2006 *Particle Swarm Optimization*. ISTE Ltd, London.

Dorigo, M. & Di Caro, G. 1999 The ant colony optimization metaheuristic. In: *New Ideas in Optimization* (Corne, D., Dorigo, M. & Glover, F. (Eds.) McGraw-Hill, London, pp 11–32.

Dorigo, M., Maniezzo, V. & Colorni, A. 1996 The ant system: optimisation by a colony of cooperating agents. *IEEE Trans. Syst. Man Cybern. Part B: Cybern.* **26**(1), 29–41.

Dorigo, M. & Stützle, T. 2004 *Ant Colony Optimization*. MIT Press, Cambridge, MA.

Eberhart, R.C. & Shi, Y. 1998 Comparison between genetic algorithms and Particle Swarm Optimization. In: Porto V.W., Saravanan N., Waagen, D. & Eiben, A.E. (eds) *Evolutionary Programming VII*, pp. 611–616. Springer.

Harbaugh, A.W., Banta, E.R., Hill, M.C. & McDonald, M.G. 2000 MODFLOW-2000, US Geological Survey Modular Ground-water Model – User Guide to Modularization Concepts and the Ground-water Process. Open File Report 00-92.

Jones, L., Willis, R. & Yeh, W.W.-G. 1987 Optimal control of nonlinear groundwater hydraulics using differential dynamic programming. *Wat. Res. Res.* **23**(11), 2097–2106.

Kennedy, J. & Eberhart, R. 1995 Particle swarm optimization. In: *Proc. IEEE International Conference on Neural Networks*. IEEE Service Center, Piscataway, NJ, pp 1942–1948.

Li, Y. & Chan Hilton, A.B. 2007 Optimal groundwater monitoring design using an ant colony optimization paradigm. *Environ. Modell. Software* **22**, 110–116.

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. *J. Wat. Res. Plann. Mngmnt.* **129**(3), 200–209.

Marryott, R.A., Dougherty, D.E. & Stollar, R.L. 1993 Optimal groundwater management. 2: Application of simulated annealing to a field scale contamination site. *Wat. Res. Res.* **29**(4), 847–860.

McKinney, D.C. & Lin, M.-D. 1994 Genetic algorithm solution of groundwater management models. *Wat. Res. Res.* **30**(6), 1897–1906.

Montalvoa, I., Izquierdoa, J., Pereza, R. & Tungb, M.M. 2008 Particle swarm optimization applied to the design of water supply systems. *Comput. Math. Appl.* **56**(3), 769–776.

Parsopoulos, K.E. & Vrahatis, M.N. 2002 Recent approaches to global optimization problems through particle swarm optimization. *Nat. Comput.* **1**(23), 235-306.

Rizzo, D.M. & Dougherty, D.E. 1996 Design optimization for multiple management period groundwater remediation. *Wat. Res. Res.* **32**(8), 2549–2561.

Wang, M. & Zheng, C. 1998 Ground water management optimization using genetic algorithms and simulated annealing: formulation and comparison. *J. AWRA* **34**(3), 519–530.

Wegley, C., Eusuff, M. & Lansey, K. 2000 Determining pump operations using particle swarm optimization. In: *Proc. Joint Conference on Water Resources Engineering and Water Resources Planning and Management, Minneapolis, MN, 30 July–2 August*. ASCE, Reston, VA, Chapter 3.

Wu, J. & Zhu, X. 2006 Using the shuffled complex evolution global optimization method to solve groundwater management models. *Lecture Note Comput. Sci.* **3841**, 986–995.

Wu, J., Zhu, X. & Liu, J. 1999 Using genetic algorithm based simulated annealing penalty function to solve groundwater management model. *Sci. China Ser. E: Technol. Sci.* **42**(5), 5.

Zheng, C. & Wang, P. P. 2003 *MGO: A Modular Groundwater Optimizer Incorporating MODFLOW and MT3DMS; Documentation and User's Guide*, University of Alabama and Groundwater Systems Research Ltd, Birmingham, AL.

First received 11 August 2009; accepted in revised form 31 January 2010. Available online 1 October 2010