Contaminant source identification using adaptive hybrid optimization of inverse groundwater transport model

Velimir V. Vesselinov[1] and Dylan R. Harp[1]

Velimir V. Vesselinov, Earth and Environmental Science Division, Los Alamos National Laboratory, Los Alamos, NM, 87544, USA. (vvv@lanl.gov)

Dylan R. Harp, Earth and Environmental Science Division, Los Alamos National Laboratory, Los Alamos, NM, 87544, USA. (dharp@lanl.gov)

[1]Earth and Environmental Science

Division, Los Alamos National Laboratory,

Los Alamos, USA.

Abstract.

A new adaptive hybrid optimization (AHO) method, called SQUADS, is proposed for solving the computationally intensive source identification problem related to contaminant transport in regional aquifers. The new method integrates an Adaptive Particle Swarm Optimization (APSO) and a Levenberg-Marquardt (LM) optimization strategy using general dynamic rules based on the runtime performance of the algorithm. The method is demonstrated on a synthetic test problem that is designed to be realistic and similar to actual contamination sites at the Los Alamos National Laboratory (LANL). The new method provides almost 100% convergence efficiency for the tested source identification problems within the allotted number of function evaluations, and substantially outperforms frequently used optimization methods such as Levenberg-Marquardt (LM), Particle Swarm Optimization (PSO), and Adaptive Particle Swarm Optimization (APSO; TRIBES). The SQUADS algorithm is applied using the code MADS.

1. Introduction

In order to assess the potential environmental risks, implement an effective remediation strategy, or design an optimal monitoring network, identification of the source location, dimensions and release history of a contaminant plume within an aquifer is beneficial in all cases and necessary in many cases. The properties of the plume source within the aquifer can be uncertain due to multiple potential sources, uncertain distribution of the contaminant at the ground surface, or uncertain transport through the vadose zone above the aquifer. It is often the case that the only available information regarding the plume source is contaminant concentrations at distributed monitoring wells. In these cases, source identification becomes an environmental forensics problem, where the goal is to identify plausible source locations, dimensions and release histories consistent with observed concentrations and estimated, assumed, or known aquifer flow and transport properties. While many source identification approaches have been presented in the literature [Dimov et al., 1996; Woodbury and Ulrych, 1996; Woodbury et al., 1998; Neupauer and Wilson, 1999; Neupauer et al., 2000; Atmadji and Bagtzoglou, 2001; Michalak and Kitanidis, 2004; Mahinthakumar and Sayeed, 2005; Neupauer et al., 2007; Dokou and Pinder, 2009], the problem remains a difficult one.

In general, two approaches exist for plume source identification: (1) solving the differential equations governing contaminant transport backwards in time commonly utilizing adjoint methods, and (2) performing a model inversion on a forward contaminant transport model. The former category of approaches can be applied to solve problems related to a single point source in a homogeneous aquifer with known properties. Examples of these

techniques include: the random walk particle method [Bagtzoglou et al., 1991], Tikhonov regularization method [Skaggs and Kabala, 1994], and adjoint method [Dimov et al., 1996; Neupauer and Wilson, 1999]. Previous approaches to solve the source identification problem by forward model inversion include linear programming and least-squares regression analysis [Gorelick et al., 1983], non-linear maximum likelihood estimation [Wagner, 1992], minimum relative entropy inversion [Woodbury and Ulrych, 1996], and geostatistically-based approaches [Michalak and Kitanidis, 2004].

Application of approaches from the latter category solve an optimization problem where model parameters associated with the source and aquifer characteristics are adjusted to match observed concentration data. The exploration of the parameter space can be computationally intensive, affected by local minima and non-linear behavior of forward model predictions with respect to model parameters. The analysis is also influenced by the discontinuous nature of predictions of contaminant concentrations at a point. For example, the predicted model concentrations at monitoring wells will be approximately zero before the arrival of the plume. This impacts the performance of gradient-based optimization techniques in particular. As a result, it is difficult to guarantee that a global minimum is achieved in the optimization process. That is why it is critical to use optimization techniques that are robust and computationally efficient in the exploration of the parameter space. Techniques to reduce the computational cost of forward model runs include: embedding the flow and transport equations directly in the optimization as binding constraints [Mahar and Datta, 1997] and using a neural network as a surrogate model [Singh et al., 2004].

We present a novel adaptive hybrid optimization (AHO) approach we call SQUADS to solve the source identification problem by performing model inversions of a forward model simulating the potential contaminant transport in the aquifer. The basis of the proposed new optimization algorithm is the coupling of global and local optimization strategies. The name SQUADS refers to the hierarchical structure of the population of solutions in the algorithm, similar to TRIBES [Clerc, Jul. 2004], but adaptively integrated with an LM optimization strategy. The benefits of combining global and local optimization strategies have been demonstrated previously on test problems and other applications [Noel and Jannett, 2004; Zhang et al., 2007; Ghaffari-Miab et al., 2007]. In fact, Mahinthakumar and Sayeed [2005] developed a step-wise hybrid approach by performing local searches on the results of a genetic algorithm for contaminant source identification. Yeh et al. [2007] introduced a hybrid approach coupling simulated annealing and tabu search to identify contaminant source location, release concentration, and release period considering a known flow field.

SQUADS utilizes an adaptive particle swarm optimization (APSO) algorithm to effectively explore the parameter space, identifying multiple promising regions, or local areas of attraction. A Levenberg-Marquardt (LM) gradient-based local search method is utilized to efficiently locate the local minimum of each of these areas. Much of the time-consuming and difficult tuning required of many optimization algorithms is reduced as the APSO algorithm does not require the specification of algorithm parameters [Clerc, 2006], and the applied LM algorithm is optimized to work well on many problems using default and internally estimated algorithm parameters [Lourakis, Jul. 2004]. The proposed new algorithm exhibits the ability to effectively traverse the complicated multi-dimensional

response surface of the problem, while efficiently locating the minimum of local areas of attraction. The new approach limits the number of necessary forward model runs by efficiently searching the parameter space for solutions with increasing consistency with observed concentrations.

We demonstrate the new approach on forward model inversions of an analytical transport model with varying degrees of freedom (i.e. variable number of free parameters). The performance of SQUADS is compared to currently available LM [Lourakis, Jul. 2004], PSO [Paricle Swarm Central, 2006], and APSO [Clerc, Jul. 2004] algorithms, demonstrating the relative benefits of the hybrid approach.

2. Particle swarm optimization

Sociobiologists have theorized that individuals within a population can benefit from the previous knowledge and experience of other members of the population while searching for sporadically distributed food sources [Wilson, 1975]. The ubiquity of schooling and flocking tendencies common among many species suggests that this is an efficient, cost-effective strategy for the survival of individuals. It is easy to recognize the analogy of organisms searching for food sources and mathematical algorithms searching for optimal solutions. This recognition led to the development of PSO by Kennedy and Eberhart [1995], building on previous research intended to graphically simulate the flocking behavior of birds. Certain aspects of the flocking behavior of this early research has been eliminated in order to improve the algorithm's performance in global optimization of mathematical functions, leading to the use of the term "swarm" to describe the graphical behavior of PSO.

The development of PSO has produced a parsimonious optimization algorithm modeling a population of initially randomly selected solutions (particles) by their position and velocity [Clerc, 2006] (the term velocity characterizes the rate of particle movement in the parameter space and does not refer to groundwater or contaminant velocity). In a D-dimensional parameter space, the position and velocity of the ith particle can be represented as $\vec{P}_i = [p_{i,1}, p_{i,2}, \ldots, p_{i,D}]$ and $\vec{V}_i = [v_{i,1}, v_{i,2}, \ldots, v_{i,D}]$, respectively. An empirical formula for determining the swarm size S has been suggested as $S = 10 + \sqrt{D}$ [Paricle Swarm Central, 2006]. Particles retain a record of the best location they have visited so far denoted as $\vec{B}_i = [b_{i,1}, b_{i,2}, \ldots, b_{i,D}]$. Particles are also informed of the best location that K other randomly chosen particles have visited, denoted as $\vec{G}_i = [g_{i,1}, g_{i,2}, \ldots, g_{i,D}]$. A standard value for K is 3 [Paricle Swarm Central, 2006]. These networks of informers are reinitialized after iterations with no improvement in the global best location of the swarm. The velocity of the ith particle in the jth dimension is updated from swarm iteration step k to $k+1$ as

$$v_{i,j}(k+1) = wv_{i,j}(k) + c_1 r_1 (b_{i,j} - p_{i,j}(k)) + c_2 r_2 (g_{i,j} - p_{i,j}(k)), \quad k = \{1, \ldots, D\}, \quad (1)$$

where w is a constant referred to as the inertia weight, c_1 and c_2 are constants referred to as acceleration coefficients, r_1 and r_2 are independent uniform random numbers in $[0, 1]$. The swarm iteration steps are also referred to as time steps because they represent the progress of swarm development in the parameter space. The parameter w controls the level of influence of the particles previous displacement on its current displacement, c_1 and c_2 scale the random influence of the particles memory and the knowledge of the particles current network of informers, respectively. A limitation on the magnitude of the

velocity V_{max} is commonly employed. The particle position at each swarm iteration step is updated as

$$p_{i,j}(k+1) = p_{i,j}(k) + v_{i,j}(k+1), \quad k = \{1, \ldots, D\}. \tag{2}$$

It has been recognized that the selection of w, c_1, c_2, and V_{max} tune the performance of PSO, modifying the balance between exploration and intensification. Manual tuning of PSO's parameters can be a delicate task. APSO algorithms have emerged in order to reduce or eliminate the often difficult and time-consuming process of parameter tuning of PSO [Cooren et al., 2009].

One of the algorithmic variants of APSO is TRIBES [Clerc, 2006], which eliminates parameter tuning and has been proven competitive on a suite of test problems with the best-known algorithms [Cooren et al., 2009]. As the name suggests, TRIBES partitions the particles into groups, referred to as "tribes", intended to facilitate the exploration of multiple areas of attraction. In this way, a hierarchical structure is established where the swarm is composed of a network of tribes, and each tribe is a network of particles. Parameter tuning is eliminated as the swarm evolves from a single tribe and the tribes evolve from single particles based on rules governing the evolution of the swarm topology and rules for generation and elimination of particles and tribes. The particle within a tribe with the lowest/highest objective function value for minimization/maximization is considered the shaman of the tribe. Information is shared only between the particles within a given tribe. Information between the tribes is shared only through the shamans. In this way, the displacement of non-shaman particles is influenced by the best particle within the tribe, while the displacement of a tribe's shamans is influenced by the best

shaman in the entire swarm. The source code for TRIBES is available from Clerc [Jul. 2004].

3. Adaptive hybrid optimization

Various approaches have been introduced to couple the global search capabilities of PSO with the efficiency of gradient-based approaches to locate local optima. Clerc [1999] introduced a PSO algorithm that adjusted particle locations based on approximations of the objective function (OF) gradient utilizing the OF values of the current particle locations. Noel and Jannett [2004] developed a hybrid PSO algorithm incorporating gradient information directly in the calculation of particle velocity. Zhang et al. [2007] coupled PSO and back-propagation to train neural networks. Ghaffari-Miab et al. [2007] developed a hybrid approach, iterating between PSO and BFGS quasi-Newton optimization. We present a hybrid approach called SQUADS that couples an APSO algorithm (modified version of TRIBES) with a Levenberg-Marquardt (LM) algorithm. The following provides a detailed description of a fine-tuned coupling of APSO and LM based on adaptive rules, where the LM optimization is applied to improve the locations of shamans (best particles within the tribes).

A flow diagram of the SQUADS algorithm is presented in Figure 3. Tables 1, 2, 3, and 4 describe the strategies and rules governing the algorithm and are indicated at the appropriate location in the flow diagram.

The algorithm is initialized similar to Standard PSO 2006 [Paricle Swarm Central, 2006] with $N_t = S = 10 + \sqrt{D}$ mono-particle tribes. The positions of the initial mono-particle tribes are determined according to rule 5 in Table 2 (refer to Table 1 for initialization

strategy selection details). The location of the first mono-particle tribe can be based on a predefined initial guess for the model parameters.

Each iteration of the algorithm is initiated by determining the informers for all particles. For non-shaman particles, this will be the shaman of their tribe (i.e. the particle with the lowest OF value within the tribe). For shaman's, this will be the shaman with the lowest OF value within the swarm, referred to as the best shaman. Particle positions are then updated according to the strategies described in Table 4. Particles are initialized to use displacement strategy 1 from Table 4.

The tribes are adapted based on whether they have demonstrated sufficient improvement in the last move. This is performed stochastically, by comparing the fraction of particles in a tribe that improved their location in the last move with a random number between 0 and 1. If the fraction is greater than the random number, the tribe is considered a "good" tribe, and the worst particle is removed from the tribe. This eliminates unnecessary function evaluations, focusing the attention of the tribe on the good particles. Otherwise, the tribe is considered a "bad" tribe, and a particle is added to the tribe (refer to Tables 1 and 2) and a randomly selected dimension of a randomly selected particle in the tribe (other than the shaman) is reinitialized randomly within the dimension/parameter's bounds. Adding a particle to a "bad" tribe is intended to increase the dispersion of the tribe.

Displacement strategies of particles are modified based on whether or not they have improved their position in the last move and if their best overall position has improved in the last move. Following the convention of Clerc [2006], we use a (+) to indicate improvement, (=) the same OF value, and (–) a worse position. The particles performance can then be denoted as one of the following: (–=), (==), (+=), and (++), where the first

symbol indicates if the particle improved its position in the last move, and second symbol indicates if the overall best position of the particle improved in the last move. Note that the best overall performance can only stay the same or improve, and an improvement in the overall performance indicates an improvement over the last position. Table 3 lists the strategy selection based on particle performance.

Following tribe adaptation, swarm adaptation will occur every $N_t * (N_t - 1)/4$ swarm iterations, unless $N_t > D$ or $10 * E > E_{max}$, in which case, the swarm adaption will occur every iterations (E is the current number of evaluations and E_{max} is the allowable number of evaluations). Each swarm adaptation adds a mono-particle tribe according to strategy 5 in Table 2 (refer to Table 1) until the maximum number of tribes is reached.

After the swarm adaptation, if $N_t > D$ or $10 * E > E_{max}$, the position of the shaman of each tribe are optimized using LM. This criteria is the same as mentioned above with respect to swarm adaptation. Therefore, once LM is being utilized, swarm adaptation occurs every iteration (refer to Figure 3). If LM is unable to reduce the OF value of the shaman's (excluding the best shaman's) position by 2/3, the status of the tribe is automatically considered "bad" for the next tribe adaptation without checking the fraction of particles that have improved in their last move (see discussion above on tribe adaptation). A particle is also added immediately according to rule 5 in Table 2.

The final step of each iteration is to perform a local random search in the empty space around each shaman [Clerc, Jul. 2004]. In this step, a random position within the largest hyperparallelepid centered on the tribe's shaman void of other particles is evaluated. If the position is an improvement over the current shaman position, the shaman is moved to this location. Otherwise, the position is forgotten.

4. Parameter space transformation

PSO and APSO algorithms are designed to operate on a bounded parameter space. The parameter ranges are predefined by the user depending on the physical constraints or prior knowledge about the parameter distributions. However, the LM optimization by default works in unbounded parameter space. There are various techniques to constrain parameter space, but typically these techniques negatively impact the LM performance. To avoid this, SQUADS operates in a transformed parameter space, where the transformed model parameter $\hat{p}$ is defined as

$$\hat{p} = \arcsin\left(\frac{p - p_{min}}{p_{max} - p_{min}} \cdot 2 - 1\right), \tag{3}$$

where p_{max} and p_{min} are the upper and lower bounds for parameter p. The APSO algorithm is performed in the transformed parameter space bounded within $[-\pi/2; \pi/2]$ in all dimensions, while the LM optimization is performed unconstrained in the transformed parameter space. Function evaluations are performed on de-transformed parameters by

$$p = p_{min} + \left(\frac{\sin(\hat{p}) + 1}{2}\right)(p_{max} - p_{min}). \tag{4}$$

In this way, the LM optimization is unaware of parameter boundaries and is unaffected by performance issues associated with calculating numerical derivatives near boundaries. It should be noted that in the process of the LM optimization, the transformed parameters can be moved outside of the $[-\pi/2; \pi/2]$ range. However, the transformed parameters are returned to equivalent values within $[-\pi/2; \pi/2]$ before being passed back to the APSO algorithm as

$$\hat{p}_{APSO} = \arcsin(\sin(\hat{p}_{LM})). \tag{5}$$

where $\hat{p}_{LM}$ represents the transformed parameters resulting from LM optimization and $\hat{p}_{APSO}$ represents the transformed parameters passed back to the APSO algorithm, thereby ensuring that APSO receives parameters within its explicitly defined, bounded parameter space.

5. Contaminant transport modeling

Contaminant transport in an aquifer can be modeled using the advection-dispersion equation for flow in the x-direction as

$$\frac{\partial c}{\partial t} = a_x u \frac{\partial^2 c}{\partial x^2} + a_y u \frac{\partial^2 c}{\partial y^2} + a_z u \frac{\partial^2 c}{\partial z^2} - u \frac{\partial c}{\partial x} - \lambda c + \frac{I}{n} \tag{6}$$

where $c(x, y, z, t)$ is the spatially (x, y, z) and temporally (t) distributed concentration, u is the pore-water velocity in the x-direction, a_x, a_y, and a_z are the longitudinal, transverse horizontal, and transverse vertical dispersivities [L], respectively, λ is the decay constant [T^{-1}], n is effective transport porosity [-], and I is the contaminant mass flux (mass per unit time) [MT^{-1}]. Equation

Based on these assumptions, an analytical solution to equation (

$$\begin{aligned} c(x,y,z,t) \;=\; & \frac{1}{8n}\int_{t_0}^{t} I(t-\tau)\exp(-\lambda\tau)\left(\operatorname{erfc}\frac{x-x_c-\frac{x_d}{2}-u\tau}{2\sqrt{a_x u\tau}} - \operatorname{erfc}\frac{x-x_c+\frac{x_d}{2}-u\tau}{2\sqrt{a_x u\tau}}\right) \\ & \times\left(\operatorname{erfc}\frac{y-y_c-\frac{y_d}{2}}{2\sqrt{a_y u\tau}} - \operatorname{erfc}\frac{y-y_c+\frac{y_d}{2}}{2\sqrt{a_y u\tau}}\right) \\ & \times\left(\operatorname{erfc}\frac{z-z_c-\frac{z_d}{2}}{2\sqrt{a_z u\tau}} - \operatorname{erfc}\frac{z-z_c+\frac{z_d}{2}}{2\sqrt{a_z u\tau}} + \operatorname{erfc}\frac{z+z_c+\frac{z_d}{2}}{2\sqrt{a_z u\tau}} - \operatorname{erfc}\frac{z+z_c-\frac{z_d}{2}}{2\sqrt{a_z u\tau}}\right) \mathrm{d}\tau \end{aligned}$$

$$-\infty < x, y < \infty, z < \infty, t > 0, \tag{7}$$

where t_0 is the time when the contaminant reaches the water table and the contaminant mass flux $I(t)$ is defined as

$$I(t) = \begin{cases} \text{if } t < t_0 & 0 \\ \text{if } t > t_0 & f \end{cases} \tag{8}$$

in order to introduce a steady contaminant mass flux f $[MT^{-1}]$ at $t > t_0$. More complicated source release functions can be also applied with this analytical solution using the principle of superposition.

To account for uncertainty in the advective-transport flow direction, model coordinates can be horizontally rotated by angle α as

$$\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{9}$$

where x_t, y_t, and z_t define the transformed coordinates. To compute concentrations over a spatial interval (e.g. a monitoring well screen), equation

Equations

It is important to note that the analytical solution assumes uniform groundwater flow and advective transport. It is well known that the contaminant transport in aquifers is substantially affected by aquifer heterogeneities [Brusseau, 1994], and the applied assumptions for development of the analytical solution may not always be valid for field applications. We use an analytical solution in our analyses as the major goal is to evaluate performance of optimization techniques to solve source identification problems. In this case, the analytical solution allows for computationally efficient execution of a large

number of model evaluations. The source identification analyses described below required up to 10,000 evaluations and the analytical solver provides about 50 model evaluations per second using a single 3.0 GHz processor.

6. Synthetic case study

A synthetic case study is utilized to test the performance of the optimization algorithms and is designed to be realistic and similar to actual contamination sites at the Los Alamos National Laboratory (LANL). Figure 2 presents a plan view of the contaminant source (black square) and monitoring well locations (15 wells). A gray-scale map of the contaminant plume at t=49 a at 1.0 m below the water table is superimposed on the figure. The well coordinates, well screen depths, and the observation time and value of contaminant concentrations are presented in Table 5. The number of concentration observations N_{obs} is 20, note that monitoring well w04 has two observations and that wells w10, w11, w12, and w13 have 2 screens each.

Table 6 lists the source, flow and transport properties describing the synthetic case and their "true" values. Maximum and minimum values based on prior information about parameter distributions are listed only for model parameters that are included in the optimization process (some of these model parameters are fixed in some of the inversion runs; details are provided below in Section 7.). The contaminant is assumed to be non-reactive ($\lambda = 0$; Table 6).

The concentration observations in Table 5 are rounded from simulated values computed in MADS obtained utilizing the "true" parameter values from Table 6. The rounding presents a more realistic precision in the observations. As a result, the "true" solution has

a non-zero objective function value equal to $10^{-0.55}$. This synthetic problem can be applied as a benchmark test case for other source identification and optimization techniques.

7. Evaluation of optimization methods

Four optimization strategies are evaluated here to solve a source identification problem: (1) LM; (2) PSO; (3) TRIBES; and (4) SQUADS. The LM strategy is an implementation of levmar [Lourakis, Jul. 2004], PSO an implementation of Standard PSO 2006 [Paricle Swarm Central, 2006], and TRIBES an implementation of code described in Clerc [2006]. LM, PSO, TRIBES and SQUADS are built into the code MADS [Vesselinov, 2010], which is utilized for the analyses presented below.

To compare the optimization methods, four optimization test cases are evaluated as listed in Table 7. All cases represent source identification problems where different sets of model parameters are assumed unknown and need to be identified by the optimization algorithm. The number of optimized parameters increases from cases A to D. Case A contains only 4 model parameters defining the lateral source location and dimensions. Case D is the most complicated, including 11 model parameters.

An analysis is presented for combinations of optimization strategies and optimization test cases, resulting in a 4 by 4 matrix of analyses. To evaluate the relative robustness of the strategies, each optimization strategy is executed 100 times for each optimization test case. The 100 optimization runs use a predefined set of random initial parameter values. The initial values are defined using Latin hypercube sample (LHS) within the parameter ranges presented in Table 6. LM and TRIBES optimization strategies are initialized from a single solution in the parameter space that is defined by the LHS initial value set. As PSO and SQUADS begin with a population of solutions, one of the solutions is set to the

parameters from the LHS initial value sample, while the remaining population of solutions are drawn from random samples according to the rules implemented by the algorithms (PSO: [Paricle Swarm Central, 2006]; SQUADS: see above). LM, PSO, TRIBES, and SQUADS are performed for up to 10,000 model evaluations (small variations within a few tens of evaluations occur for different runs due to differences in the termination criteria implemented by each method). LM never required 10,000 model evaluations because of the other termination criteria. The following criteria are defined by default in levmar [Lourakis, Jul. 2004]: (1) the maximum change in any parameter is less than 10^{-5}; (2) the relative change in the L2 norm of the change in the parameter values is less than 10^{-5} of the L2 norm of the parameter values; (3) the OF reaches a value of zero; (4) the Jacobian is close to singular, and (5) maximum number of LM iterations is achieved. In the analyses presented below, typically, the LM terminated due to either criteria (1) and (2) by 1300 model evaluations. Even if the LM termination criteria were modified to allow for optimization to continue until 10,000 evaluations, this would not have resolved the problem in avoiding local minima to achieve the global minimum.

In all four optimization strategies, a sum of the squared residuals (SSR) functional form is used for the OF as

$$\Phi(\theta) = \sum_{i=1}^{N} (\hat{y}_i(\theta) - y_i), \tag{10}$$

where Φ is the OF, θ is a vector containing the optimization parameters, $\hat{y}_i(\theta)$ is the ith simulated concentration using parameter values in θ, and y_i is the ith calibration target (observed concentration). $\hat{y}_i(\theta)$ is computed as the average of the simulated concentrations

306 at the top and bottom of the screen (screen top and bottom depths are tabulated in
307 Table 5).

8. Results and discussion

308 Figure 3 presents a matrix of OF histograms for each of the analysis combinations.
309 Rows denote the optimization test case (refer to Table 7) and the columns denote the
310 optimization strategy. Moving down the rows of the matrix present analyses with lower
311 degrees of freedom. Complimentary plots of OF versus number of function evaluations
312 are presented in Figure 4. Due to rounding of the concentrations, the "true" solution is
313 achieved at an OF value of $10^{-0.55}$. Table 8 presents the probability of each optimization
314 strategy attaining the "true" solution for each case.

315 By inspecting Figures 3 and 4 as well as Table 8, several conclusions can be drawn as
316 discussed below.

317 PSO and TRIBES have similar overall performance. TRIBES fails to locate the min-
318 imum in all the cases. PSO fails to locate the global minimum for cases B, C, and D,
319 but manages to identify the global minimum with low probability (0.12) in the simplest
320 case (A). PSO and TRIBES performance generally deteriorates with increasing problem
321 complexity (from case A to case D).

322 LM is generally performing better than PSO and TRIBES, but it is less robust than
323 SQUADS. LM identifies the accurate solution with probability ranging from 0.13 to 0.58
324 (Table 8). The LM performance in the simplest case (A) is clearly affected by three local
325 minima (Figure 3) with OF values approximately equal to $10^{2.8}$, $10^{5.2}$, and $10^{6.1}$. Local
326 minima affect its performance in the other 3 cases as well (B, C and D; Figure 3). For cases

B, C and D, the LM performance consistently declines as the dimension of the parameter space increases.

It is somewhat surprising that the LM performance is better in case B than in the simpler case A. Analysis of the optimization results demonstrate that in case B, the performance is enhanced because LM is allowed to increase a_x, a_y, and a_z above their optimal values during the early optimization iterations. The higher dispersivity values increase the size of the plume and the observed concentrations at the monitoring wells far from the source. This makes the optimization problem more computationally amenable for gradient-based methods such as LM where derivatives of model predictions with respect to model parameters are computed.

SQUADS demonstrates the highest robustness in identifying the "true" solution of the source identification problem for all the cases (A, B, C, and D). In case D, SQUADS did not achieve the global minimum in all the runs within limiting number of model evaluations. The solutions that did not reach the global minimum are still in very close vicinity to it (Figures 3 and 4). By inspecting Figure 4, it appears possible that the global minimum would be achieved with a marginal increase in the number of function evaluations. Overall, SQUADS demonstrates the ability to converge at less than 10,000 model evaluations. In cases A, B and C, SQUADS converged within 1800, 3400 and 5800 evaluations, respectively; in case D, SQUADS converged for less than 6000 evaluations in most of the cases (Figures 4).

9. Conclusions

A new adaptive hybrid optimization method called SQUADS is proposed for solving the computationally intensive source identification problem related to contaminant trans-

port in regional aquifers. The method is tested to solve synthetic test problems that are designed to be realistic and similar to actual real-world problems [Los Alamos National Laboratory, 2008]. The new method provides almost 100% convergence efficiency for the tested source identification problems. SQUADS substantially outperforms frequently used optimization methods such as LM, PSO, and TRIBES. The application of the SQUADS algorithm is performed using the code MADS [Vesselinov, 2010]. MADS can be executed in forward mode, in which case it will produce model predictions of concentrations at the monitoring wells based on provided model parameters. This allows the code to be coupled with external optimization algorithms. MADS can also be executed in an inverse mode, optimizing model parameters based on provided concentrations at the monitoring wells using the internal optimization strategies, such as LM, PSO, TRIBES and SQUADS. MADS and other files needed to execute the synthetic problem are available at http://www.ees.lanl.gov/staff/monty/codes/mads.html

References

Atmadji, J., Bagtzoglou, A. C., 2001. State of the art report on mathematical methods for groundwater pollution source identification. Environmental Forensics 2, 205–214.

Bagtzoglou, A., Tompson, A., Dougherty, D., 1991. Probabilistic simulation for reliable solute source identification in heterogeneous porous media. Water Resources Engineering Risk Assessment. NATO ASI Series G29, 189–201.

Brusseau, M. L., 1994. Transport of reactive contaminants in heterogeneous porous media. Reviews of Geophysics 32 (3), 285–313.

Clerc, M., 1999. The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization. IEEE, 1951–1957.

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

Clerc, M., Jul. 2004. Tribes, an adaptive particle swarm optimization algorithm in C. [web page] http://clerc.maurice.free.fr/pso/Tribes/TRIBES-D.zip, [Accessed on 4 Nov. 2010.].

Cooren, Y., Clerc, M., Siarry, P., 2009. Performance evaluation of TRIBES, an adaptive particle swarm optimization algorithm. Swarm Intelligence 3, 149–178.

Dimov, I., Jaekel, U., Vereecken, H., 1996. A numerical approach for determination of sources in transport equations. Computers Math. Applic. 32 (5), 31–42.

Dokou, Z., Pinder, G. F., 2009. Optimal search strategy for the definition of a dnapl source. Journal of Hydrology 376, 542–556.

Ghaffari-Miab, M., Farmahini-Farahani, A., Faraji-Dana, R., Lucas, C., 2007. An efficient hybrid swarm intelligence-gradient optimization method for complex time Grees's functions of multilayer media. Progress in Electromagnetics Research 77, 181–192.

Gorelick, S., Evans, B., Remson, I., 1983. Identifying source of groundwater pollution: An optimization approach. Water Resources Research 19 (3), 779–790.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proceedings of the IEEE International Conference on Neural Networks. IEEE Press, Piscataway, pp. 1942–2948.

Los Alamos National Laboratory, 2008. Fate and transport investigations update for chromium contamination from Sandia Canyon. Tech. rep., Environmental Programs Directorate, LANL, Los Alamos, New Mexico, lA-UR-08-4702.

Lourakis, M., Jul. 2004. levmar: Levenberg-marquardt nonlinear least squares algorithms in C/C++. [web page] http://www.ics.forth.gr/~lourakis/levmar/, [Accessed on 4 Nov. 2010.].

Mahar, P., Datta, B., 1997. Optimal monitoring network and ground-water-pollution source identification. Journal of Water Resources Planning and Management - ASCE 123 (4), 199–207.

Mahinthakumar, G., Sayeed, M., 2005. Hybrid genetic algorithm – local search methods for solving groundwater source identification inverse problems. Journal of Water Resources Planning and Management - ASCE 131 (1), 45–57.

Michalak, A. M., Kitanidis, P. K., 2004. Estimation of historical groundwater contaminant distribution using the adjoint state method applied to geostatistical inverse modeling. Water Resources Research 40.

Neupauer, R. M., Borchers, B., Wilson, J. L., September 2000. Comparison of inverse methods for reconstructing the release history of a groundwater contamination source. Water Resources Research 36 (9), 2469–2475.

Neupauer, R. M., Lin, R., O'Shea, H., 2007. Conditional backward probability modeling to identify sources of groundwater contaminants subject to sorption and decay. Water Resources Research 43.

Neupauer, R. M., Wilson, J., 1999. Adjoint method for obtaining backward-in-time location and travel time probabilities of a conservative groundwater contamination. Water Resources Research 35 (11), 3389–3398.

Noel, M. M., Jannett, T. C., 2004. Simulation of a new hybrid particle swarm optimization algorithm. IEEE, 150–153.

Paricle Swarm Central, 2006. http://www.particleswarm.info/Standard_PSO_2006.c.

Singh, R., Datta, B., Jain, A., 2004. Identification of unknown groundwater pollution sources using artificial neural networks. Journal of Water Resources Planning and Man-

agement - ASCE 130 (6), 506–514.

Skaggs, T., Kabala, Z., 1994. Recovering the release history of a groundwater contaminant. Water Resources Research 30 (1), 71–79.

Vesselinov, V. V., 2010. Mads, modeling analysis and decision support toolkit in C. [web page] http://www.ees.lanl.gov/staff/monty/codes/mads.html, [Accessed on 15 Nov. 2010.].

Wagner, B., 1992. Simultaneous parameter estimation and contaminant source characterization for coupled groundwater flow and contaminant transport modeling. Journal of Hydrology 135 ((1-4)), 275–303.

Wilson, E., 1975. Sociobiology: The new synthesis. Belknap Press, Cambridge, MA.

Woodbury, A., Sudicky, E., Ulrych, T., Ludwig, R., July 1998. Three-dimensional plume source reconstruction using minimum relative entropy inversion. Contaminant Hydrology 32 (1), 131–158.

Woodbury, A. D., Ulrych, T. J., September 1996. Minimum relative entropy inversion: Theory and application to recovering the release history of a groundwater contaminant. Water Resources Research 32 (9), 2671–2681.

Yeh, H.-D., Chang, T.-H., Lin, Y.-C., 2007. Groundwater contaminant source identification by a hybrid heuristic approach. Water Resources Research 43.

Zhang, J.-R., Zhang, J., Lok, T.-M., Lyu, M. R., 2007. A hybrid particle swarm optimization-back-propagation algorithm for feedforward neural network training. Applied Mathematics and Computation 185, 1026–1037.

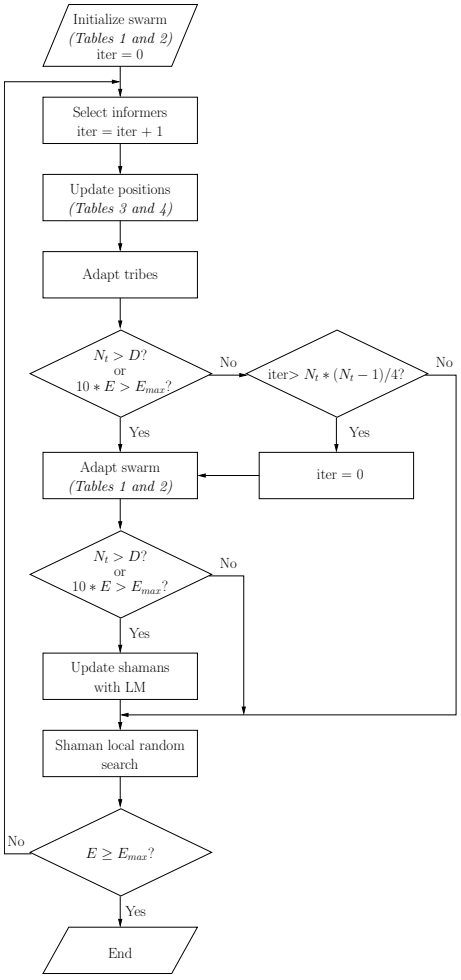

Figure 1. Flow diagram of SQUADS. Italicized notations of tables indicate the table describing rules governing the behavior of the algorithm at that location in the algorithm. *iter* is an index representing the current swarm iteration step, N_t is the current number of tribes, D is the dimension of the parameter space, E is the current number of function evaluations, E_{max} is the allowable number of function evaluations, and N_p is the current number of particles.

Table 1. Particle initialization strategy (refer to Table 2) within AHO algorithm.

	Rule selection
First particle of the algorithm	1
If initial population is greater than 1, other initial particles	5
Particle added to "bad" tribe (tribe adaptation)	randomly choose between 2 and 5
Mono-particle tribe added (swarm adaptation)	5
LM unable to reduce OF of shaman by 2/3	5

Table 2. Rules governing the initialization of a particle's location. U(a,b) is a uniform distribution with maximum b and minimum a

1. User specified

2. Randomly chosen position within parameter space:

p_{new_j} =U(p_{min_j}, p_{max_j}), $\quad j = 1, \ldots, D$

3. Randomly chosen within hyperparallelepid surrounding the best position of the swarm with dimensions $(2 \cdot r_j)$ determined by Euclidean distance between the swarm's and tribe's best position:

$r_j = |p_{best_j} - p_{tribe\ best_j}| \quad j = 1, \ldots, D$

p_{new_j} =U$(p_{best_j} - r_j, p_{best_j} + r_j) j = 1, \ldots, D$

4. On one of the vertices of the parameter space with equal probability of being the max or min of each dimension:

if (U$(0, 1) < 0.5$) then $p_{new_j} = p_{min_j}$, else $p_{new_j} = p_{max_j} \quad j = 1, \ldots, D$

5. Randomly chosen within the largest empty hyperparallelepid of the parameter space

Table 3. Particle strategy selection based on performance. Refer to Table 3 for strategy definitions.

Particle status	Strategy selection
(–=)	randomly choose any strategy other than current one
(==)	randomly choose between strategy 2 and 3
(+=),(++)	change to strategy 1 with 50% probability

Table 4. Particle displacement rules. $\mathrm{N}(\mu, \sigma)$ is a normal distribution with a mean μ and standard deviation σ, $f(-)$ is the value of the objective function, $\vec{g} = [g_1, g_2, \ldots, g_D]$ is the location of the particles designated informer and $\vec{B} = [b_1, b_2, \ldots, b_D]$ the particle's current best location.

1. $p_j = \mathrm{N}(b_j, |b_j - g_j|) \quad j = 1, \ldots, D$
2. $p_j = c_1 \cdot \mathrm{N}(b_j, |b_j - g_j|) + c_2 \cdot \mathrm{N}(g_j, |b_j - g_j|)$
3. $p_j = [c_1 \cdot N(b_j, |b_j - g_j|) + c_2 \cdot N(g_j, |b_j - g_j|)] \cdot \left[1 + N\left(0, \frac{f(\vec{g}) - f(\vec{p})}{f(\vec{g}) + f(\vec{p})}\right)\right]$

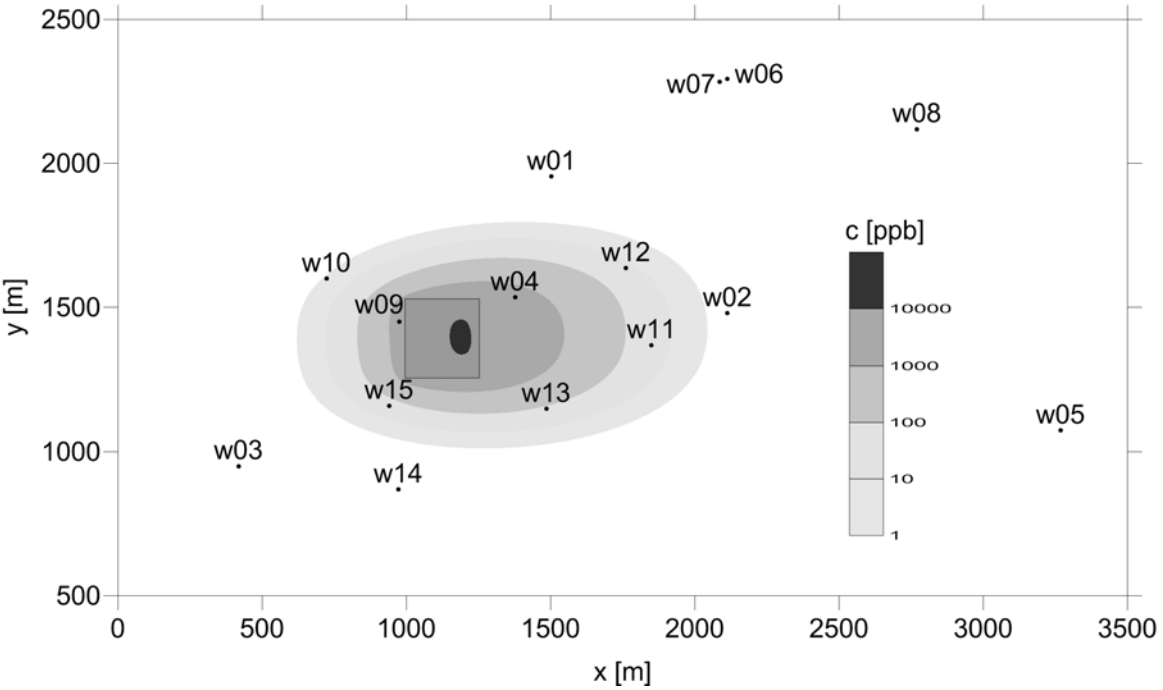

Figure 2. Plan view of contaminant source (black square) and well locations. The gray-scale map of the contaminant plume at t=49 a at 1.0 m below the water table is superimposed on the map.

Table 5. Well coordinates, screen top (z_{top}) and bottom (z_{bot}) depths below the water table, and year and value of contaminant concentration observations

Well	x [m]	y [m]	z_{top} [m]	z_{bot} [m]	t [a]	c [ppb]
w01	1503	1954	5.57	12.55	49	0
w02	2113	1479	36.73	55.14	49	0
w03	418	950	0	15.04	49	0
w04	1377	1534	13.15	20.41	44	350
					49	432
w05	3268	1074	26.73	33.71	49	0
w06	2112	2294	69.01	83.98	49	0
w07	2086	2284	11.15	18.19	49	0
w08	2770	2119	4.86	11.87	49	0
w09	975	1450	3.66	10.09	49	981
w10	723	1599	3.32	9.63	49	1.1
			23.2	26.24	49	0.1
w11	1850	1368	4.94	7.99	49	22
			32.46	35.48	49	0.3
w12	1761	1636	3.59	6.64	49	15
			32.51	38.61	49	0.17
w13	1485	1149	3	6	50	72
			36	42	50	0.26
w14	972	869	3	6	50	0
w15	940	1160	3	6	50	38

Table 6. Aquifer and contaminant source properties defining a synthetic case. Intervals are listed for parameters included in the optimization process. Minimum and maximum values are omitted for fixed parameters in all the analyses.

Property	Parameter	True	Min	Max
Source center [m]	x_c	1124	210	1460
	y_c	1393	1230	1930
	z_c	0.5	–	–
Source dimension [m]	x_d	258	1	400
	y_d	273	1	500
	z_d	1.0	–	–
Contaminant flux [kg/a]	I	16	0.01	100
Porosity [m/m]	n	0.1	–	–
Decay constant [1/a]	λ	0	–	–
Start time [a]	t_0	0	0	43
Flow angle [degrees]	α	3	-20	20
Contaminant pore velocity [m/a]	u	5	0.01	200
Dispersivity [m]	a_x	70	10	140
	a_y	15	1	30
	a_z	0.3	0.1	1

Table 7. Four optimization test cases with increasing level of complexity where D is the number of optimization parameters, DOF is the degrees of freedom of the optimization, and the number of observations N_{obs} in each case is 20.

Case	D	DOF	Parameters
A	4	16	$x_c,\ y_c,\ x_d,\ y_d$
B	7	13	$x_c,\ y_c,\ x_d,\ y_d,\ a_x,\ a_y,\ a_z$
C	8	12	$x_d,\ y_d,\ I,\ \alpha,\ u,\ a_x,\ a_y,\ a_z$
D	11	9	$x_c,\ y_c,\ x_d,\ y_d,\ I,\ t_0,\ \alpha,\ u,\ a_x,\ a_y,\ a_z$

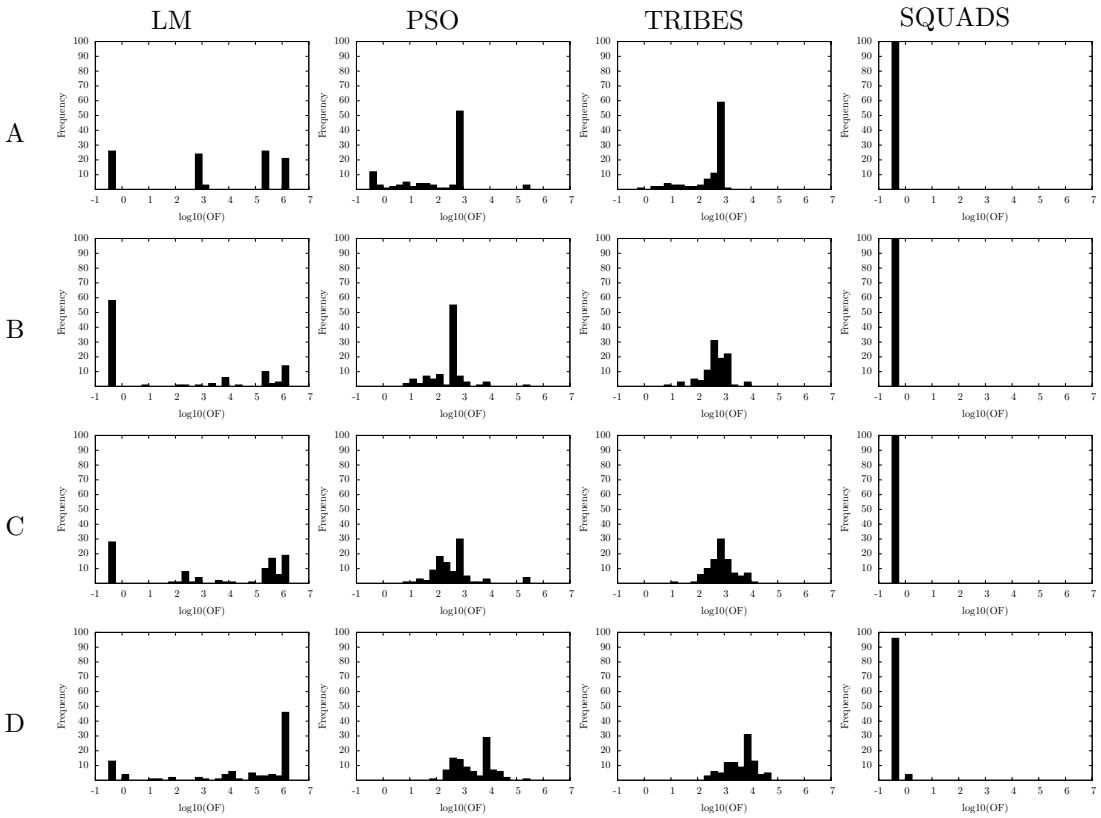

Figure 3. Matrix of OF histograms. Rows denote the optimization case (Table7) and columns denote the optimization strategy of each analysis.

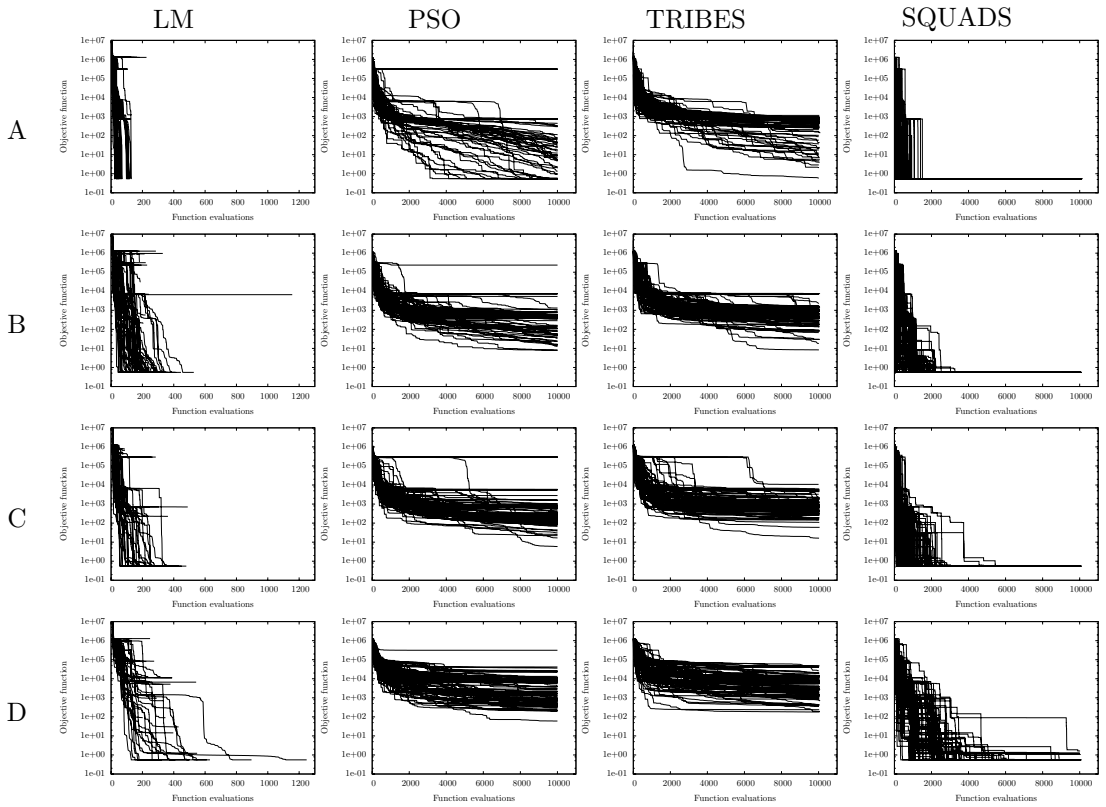

Figure 4. Matrix of plots of OF versus number of evaluations. Rows denote the optimization case (Table7) and columns denote the optimization strategy of each analysis. Note that LM plots use a different abscissa axis truncated at 1300 function evaluations.

Table 8. Probability of reaching the "true" solution.

Case	LM	PSO	TRIBES	SQUADS
A	0.26	0.12	0.00	1.00
B	0.58	0.00	0.00	1.00
C	0.28	0.00	0.00	1.00
D	0.13	0.00	0.00	0.96