Algorithmic funnel-and-gate system design optimization

Claudius M. Bürger,[1] Peter Bayer,[1] and Michael Finkel[1]

Received 23 March 2006; revised 27 April 2007; accepted 6 June 2007; published 24 August 2007.

[1] Funnel-and-gate systems (FGSs), which constitute a common variant of permeable reactive barriers used for in situ treatment of groundwater, pose particular challenges to the task of design optimization. Because of the complex interplay of funnels and gates, the evolutionary algorithms applied have to cope with multimodality, nonseparability, and nonlinearity of the optimization task. We analyze these features in a test case, introducing an objective function for design cost and constraints to account for plume capture and detention time in the gate reactors. We show that the derandomized evolution strategy with covariance matrix adaptation (CMA-ES) does solve the given design optimization problem with high success rates. We further examine the performance of the algorithm for the example of four-gate systems in three heterogeneous template aquifers. Here a special focus is set on the parameterization of the FGS (i.e., the problem encoding). The comparison of three different encodings reveals their significance concerning the search progress and its success. Among the found optimal and near-optimal design solutions, mutual patterns were recognized. In particular, a large central barrier seems to be a superior feature.

Citation: Bürger, C. M., P. Bayer, and M. Finkel (2007), Algorithmic funnel-and-gate system design optimization, *Water Resour. Res.*, *43*, W08426, doi:10.1029/2006WR005058.

1. Introduction

[2] Permeable reactive barriers (PRBs) have evolved as a promising alternative to traditional contaminant plume management methods, such as pump-and-treat (P&T) systems [e.g., *Gavaskar*, 1999]. Unlike the well-established P&T approach, which is based on active hydraulic containment by wells, PRBs are essentially passive technologies. Here vertical barriers are installed in an aquifer to transect a contaminant plume and prevent it from further downgradient spreading. These barriers can be constructed as continuous permeable walls, which are uniformly fabricated out of reactive media such as zero-valent iron to degrade contaminants in situ. Alternatively, reactive barrier elements or zones (''gates'') are intersected by impermeable segments (''funnels'') which direct the plume to the gates (Figure 1). This specialized variant of PRBs, the so-called funnel-and-gate system (FGS), provides a more spatially focused in situ treatment of contaminants [*Starr and Cherry*, 1994; *Teutsch et al.*, 1997]. This could be an advantage in highly heterogeneous aquifers or in the case of complex contaminant mixtures, which have to be treated in multiple reactors. Furthermore, if the longevity of the treatment material used cannot be assured or, as is common for applications with sorptive media, has only a limited lifetime, it is sufficient to only exchange exhausted media in the gates, instead of replacing the entire wall [e.g., *U.S. Environmental Protection Agency*, 2002; *Bayer and Finkel*, 2006a]. Compared with continuous walls, however, the appropriate design of FGSs is more challenging. In addition to the decision of the total length, width, and geometry of a continuous wall, an FGS design requires an arrangement of the sequence of funnels and gates according to site-specific conditions. Both the various degrees of freedom in technology design, as well as the relevant site-specific factors, such as the hydraulic situation, the distribution of contaminants, and the desired cleanup goals, can result in a demanding optimization problem.

[3] Despite the fact that PRBs and P&T remediation technologies are quite different conceptually, the overall management problem is essentially the same: A contaminant plume has to be captured and contaminants have to be removed from the subsurface system. Potentially, other (additional) constraints, such as a lack of severe decrease or increase of the water table or restricted space for technology implementation, have to be taken into account. An optimal solution would achieve all that at minimum costs. If a calibrated site model is available, the search for the most cost-efficient design may be automated by a mathematical formulation with the management problem as an objective function (OF). Usually, the OF computes the costs of the technology as a function of design parameters, which are minimized by the use of an appropriate optimization algorithm. The algorithm then suggests adjustments to formal design parameters, so that a cost-optimal design for a specified number of gate reactors is found.

[4] In the scientific literature the P&T technology appears to have been the main focus of algorithmic optimization efforts for more than 2 decades. Of the various different mathematical techniques applied (e.g., *Atwood and Gorelick* [1985] (linear programming with response matrix); *Lefkoff and Gorelick* [1986] (quadratic programming); *Ratzlaff et al.* [1992] (mixed-integer programming); *Culver and Shoemaker* [1993] (differential dynamic programming); *Dougherty and Marryott* [1991]

[1]Center for Applied Geoscience, University of Tübingen, Tübingen, Germany.

0043-1397/07/2006WR005058$09.00

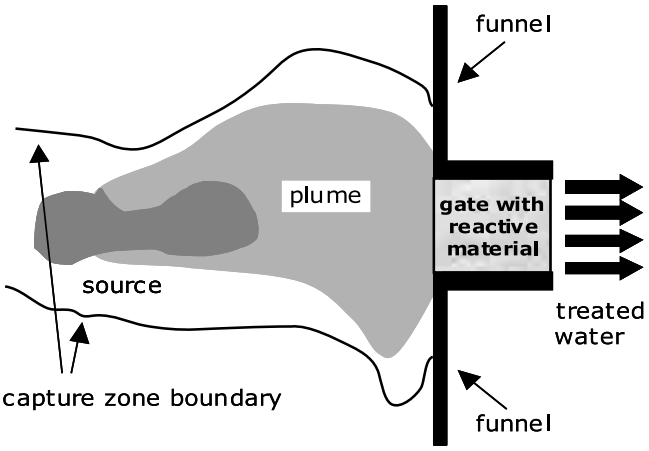

Figure 1. Sketch of the funnel-and-gate system (FGS) concept.

(simulated annealing); *Wang and Ahlfeld* [1994] (nonlinear programming); *Karatzas and Pinder* [1993] and *Papadopoulou et al.* [2003] (outer approximation)), heuristic methods seem to have recently become more and more popular for solving water resources optimization problems (e.g., *Zheng and Wang* [1999] (tabu search); *Zheng and Wang* [2002] (evolutionary algorithms with response function); *Nunes et al.* [2004] (simulated annealing); *Bayer and Finkel* [2004] (evolution strategy and genetic algorithm (GA)), with the manifold GA variants being the most popular (e.g., *Guan and Aral* [1999] (progressive GA); *Reed et al.* [2000] (simple GA); *Hsiao and Chang* [2005] (combination of GA and constraint differential dynamic programming)). Nevertheless, it seems that PRB design problems have rarely been dealt with by using rigorous mathematical optimization methods. Recent mathematical work on the topic involves uncertainty and data worth analyses of FGSs [*Cirpka et al.*, 2004] or analytical capture zone expressions for continuous barriers [*Craig et al.*, 2006].

[5] A burdensome aspect of the FGS design optimization is that plume capture can only be achieved through the intricate interplay between funnels and gates. It is therefore expected to face a nonseparable optimization problem, in which individual design parameters show a strong correlation. Furthermore, considering heterogeneous aquifers in particular, contaminant plumes are likely to be distributed among several high-conductivity zones, which can result in a multimodal optimization problem (i.e., more than one local minimum existing). Similar to formulations for well placement of P&T systems in a heterogeneous aquifer [*Zheng and Wang*, 1999; *Bayer and Finkel*, 2004], nonlinearity of the FGS optimization problem is expected. Adjusting funnel and/or gate dimensions may introduce a linkage or blockage of high- or low-conductivity zones, such that the groundwater flow is not likely to follow a linear relationship. For example, doubling the system width may not just double the capture zone, but also may result in shifts and/or distortions of varying severity.

[6] The nonlinearity, nonseparability, and multimodality of the FGS optimization problem as well as the fact that a numerical computer model is involved in the computation of OF values pose a particular challenge to the optimization task. Classical optimization techniques based on gradients (steepest descent, conjugate gradients) or higher-order derivatives (Newton-methods) are too computationally intense and place high demands on the smoothness of the OF, which may not be satisfiable due to the required discretization of the numerical model.

[7] In order to find an optimal FGS design, we theoretically have to deal with positions, funnel widths, and gate widths as continuous variables (Figure 1). However, as mentioned above, the necessary spatial discretization of groundwater models renders these variables representable only as discrete model cells. Hence, unless adaptive grid refining [e.g., *Bayer and Finkel*, 2006b] is a standard component of the modeling software, the search space can only be effectively explored in discrete steps. Therefore we face a discrete optimization problem in this case.

[8] Facing the challenges stated above, we decided to approach the FGS design problem with variants of the general class of evolutionary algorithms. These represent iterative stochastic optimization algorithms, which use heuristics based on biological evolution principles to guide the search procedure: From a population of OF evaluation points (“elders”), a subpopulation is selected according to their fitness (i.e., OF value) to create an offspring population by recombination and mutation (i.e., a further modification of the selected elder points). Principally, a single new evaluation point (“individual”) is not expected to have a better fitness than its elders. Improvement is only achieved by a selection from the (whole) new set of evaluation points (“offspring”). Typically, the selection process itself is a mere ranking of individuals by their fitness. These characteristics make evolutionary algorithms very robust compared with classical optimization algorithms, in terms of handling nonsmoothness and the presence of noise. Owing to their stochastic nature, though, convergence properties can only be stated in a probabilistic sense.

[9] For clarification, a few technical terms are subsequently introduced. The individual parameters describing the design of the FGS are called design parameters (e.g., the width of a gate opening). Specific parameterizations of these developed for algorithmic purposes are termed an (optimization) problem encoding. Accordingly, the encoding parameters are the OF parameters or decision variables. One complete set of decision variable values is referred to as an evaluation point or individual. The OF may also be called fitness function. Each algorithm restart defines an optimization run (OR). More specifically, this means running the algorithm with a specified population size for a set number of generations where its initial population or starting point is determined randomly based on a user-specified seed. An algorithm is evaluated by its ability to find optimal or near-optimal solutions (FGS designs) for a possibly small number of OF evaluations (*nfeval*).

[10] The most well known variant of evolutionary algorithms are probably binary-coded GAs. Operating in a discrete search space and being regarded as a (heuristic) global optimization procedure, these GAs appear to be an obvious choice for the FGS design problem. Because of their popularity and abundant use, e.g., in P&T optimization, we will not go into much detail regarding GAs in this study. However, a short outline of the working principles

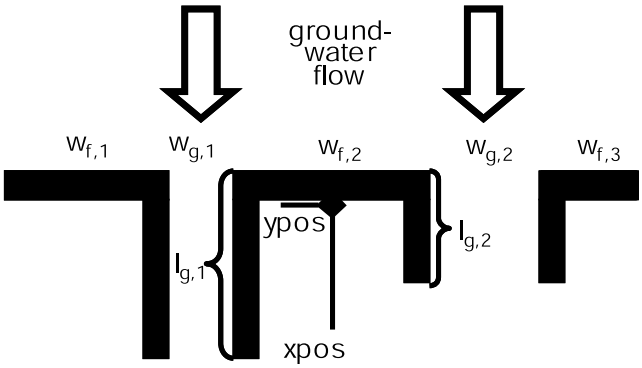

Figure 2. Design variables used to describe funnel-and-gate systems.

and the applied parameterization is provided further below. We refer interested readers to *Goldberg* [1989] or *Reed et al.* [2000].

[11] As a further option we explore a less well known evolutionary algorithm: a real-valued (derandomized) evolution strategy. Its choice was inspired by the promising results of *Yoon and Shoemaker* [1999] and *Bayer and Finkel* [2004] for bioremediation and P&T optimization, respectively. More specifically, the evolution strategy with covariance matrix adaptation (CMA-ES) [*Hansen and Ostermeier*, 2001; *Hansen et al.*, 2003] is used here. An evolution strategy, like a GA, works derivative-free. However, in contrast to a GA, the direction of the search is based on the sequence of selected individuals, so that an evolution strategy is indeed following a kind of overall ''gradient.''

[12] A distinctive feature of the CMA-ES is that for every generation it updates a multidimensional normal distribution from which the next population is sampled. This normal distribution is given by its mean and covariance matrix $\mathbf{C}$. The latter models decision variable correlations. From a geometric perspective, the covariance matrix can be viewed as a hyperellipsoid whose surface defines an equal probability density for the population. The eigenvalues of the covariance matrix are the squared lengths of the principle axes of the hyperellipsoid, and the eigenvectors correspond to the principle axes [see *Hansen and Ostermeier*, 2001]. One general design principle of the CMA-ES is to think of the OF locally as a convex-quadratic function $f_{\mathbf{H}}(\mathbf{x}) = \frac{1}{2} \times \mathbf{x}^t\mathbf{H}\mathbf{x}$, where the Hessian matrix $\mathbf{H}$ is a positive definite matrix and superscript t denotes transpose. There exists a close relation between $\mathbf{H}$ and $\mathbf{C}$: Setting $\mathbf{C} = \mathbf{H}^{-1}$ on $f_{\mathbf{H}}$ is equivalent to optimizing the isotropic function $f_{\text{sphere}}(\mathbf{x}) = \frac{1}{2} \times \mathbf{x}^t\mathbf{x} = \frac{1}{2} \times \Sigma_i\, x_i$ (where $\mathbf{H} = \mathbf{I}$, with $\mathbf{I}$ being the corresponding unit matrix) with $\mathbf{C} = \mathbf{I}$. This means that on convex-quadratic OFs, setting the covariance matrix of the search distribution to the inverse Hessian is equivalent to rescaling the ellipsoid function (i.e., any of those $f_{\mathbf{H}}$ introduced above) into a spherical one. Consequently, it is assumed that the optimal covariance matrix approximates the inverse Hessian matrix up to a constant factor (see http://www.bionik.tu-berlin.de/user/niko/cmatutorial.pdf, p. 7).

[13] As mentioned above the CMA-ES is a real-valued algorithm and operates in a real-valued search space. Naturally, this raises the question of whether the CMA-ES is a suitable algorithm for the discrete FGS design optimization problem. A necessary condition for the applicability of evolution strategies in general was given by *Rechenberg* [1994] and is known as the principle of strong causality: Small changes in decision variable values should cause small changes in the OF value. Although this is related to the mathematical notion of continuity, this principle is weaker in the sense that only a fitness-distance correlation is demanded. Therefore absence of continuity does not limit the application of the CMA-ES a priori (e.g., see applications of evolution strategies by *Herdy* [1991] or *Cai and Thierauf* [1996]).

[14] As a partial differential equation is solved in the flow problem, we would not initially question strong causality for the FGS design problem. However, sources of disturbance arise from heterogeneous (discretized) hydraulic conductivity fields, in particular, the strong discontinuity of the FGS itself and the way plume capture is incorporated in the design problem. Furthermore, we would also expect the model discretization to be influential, at least in the vicinity of an optimum. There the presence of a 'local gradient' determines whether the optimum is detectable for the CMA-ES or not. If the discretization is too coarse, local minima, defined only by their nearest neighboring evaluation points, are likely to be stepped over.

[15] This article is organized as follows. After a detailed description of the modeling approach, the problem formulation, and an outline of the applied evolutionary algorithms, we first explore a simplified one-gate FGS test case in terms of the overall shape of the fitness landscape of the FGS design problem. For the test case, also the global optimum is determined by complete enumeration. This allows for a subsequent investigation of the reliability of convergence of the evolutionary algorithms under consideration toward the global optimum.

[16] In a second part, optimal four-gate FGSs are sought for three aquifer scenarios, which are characterized by different, heterogeneous hydraulic conductivity fields. In particular, the potential for an increase in speed of the optimization process is investigated. Emphasis is given to the comparison of three newly developed problem encodings, that is, finding design parameterizations which are most favorable with respect to the search principles of the applied optimization algorithm. Finally, the obtained optimal and near-optimal designs are visually inspected in order to find mutual patterns of optimality.

2. Method

2.1. FGS Design and Cost Model

[17] Within the scope of this first study on algorithmic FGS design optimization with individual gate and funnel widths as decision variables, we chose a number of simplifying assumptions concerning the design description and the underlying cost model. The adaptation of the FGS design is limited to straight FGSs [cf. *Sedivy et al.*, 1999], meaning that all funnel segments are positioned on a straight line. All FGSs are modeled as fully penetrating the aquifer and as being keyed to an underlying aquitard. Parameters identifying a particular design are the overall position (usually represented by an (x_{pos}, y_{pos}) coordinate pair); the number of gates n_g; the individual n_g gate widths $w_{g,i}$ as well as the $n_g + 1$ funnel widths $w_{f,i}$. Figure 2 illustrates these parameters for $n_g = 2$.

[18] A typical task for a practitioner involves ensuring the minimization of project costs for a remediation technology, and also compliance with remediation goals, for example, plume control or aquifer decontamination. Although cost elements for remediation technologies can be distinguished at any level of detail, many may be site-specific. To keep to the general aim of this study, only the basic elements of the FGS design are included in the cost function, i.e., the funnel segments and the gate segments. This study focuses predominantly on static plume control rather than analyzing the long-term cleanup process. This means that valid FGS variants are all those which (1) achieve hydraulic capture and (2) guarantee sufficient contaminant degradation (or retention) in the gates.

[19] The chemical processes within the in situ gate reactors are incorporated by a specified detention time, t_{det}, which is assumed to be required in order to properly treat the contaminants. This formulation achieves independence from contaminant concentrations to avoid the computational burden of advective-dispersive transport calculations. Depending on the applied unit costs, the gate may also be viewed as a high-permeability reactive zone, where nutrients, oxidants, reducing agents, or microorganisms may be added for enhanced natural attenuation. Excluding the added substances from cost considerations, this means that no material change is anticipated throughout project duration, and that only capital costs have to be taken into account. Cost elements which are not exclusively dependent on an individual FGS design (e.g., licensing, labor), and/or which would, in good approximation, only add the same constant amount to the cost function (e.g., site installation costs), are also deliberately excluded.

[20] The assumptions listed above represent a simplification of the real system which deserves further discussion. The consideration of contaminant type specific degradation reactions, kinetics, and concentration distributions may lead to different reactor requirements for different funnel and gate combinations. However, the cost-driving outcome of a reactor simulation would still be a required detention time. This detention time would then no longer be constant, but dependent on the decision variables. In contrast, one of the intended functions of an FGS is, indeed, to "funnel" the flow paths to a defined reactive zone in order to provide homogenized and averaged contaminant concentrations. In view of this, and assuming constant properties of the reactive material, we believe neglecting the functional dependency of the required detention time on the decision variables can be justified for the scope of this study.

[21] The incorporation of time-dependent decisions, e.g., when to add nutrients, apply reactivation measures, does require process simulations, e.g., of enhanced natural attenuation, reactive material deactivation, or pore clogging over time. Accurate simulations are, if at all available, very computationally demanding [e.g., *Maier and Grathwohl*, 2006] and hence prohibitive for use within this elementary optimization study. Costs arising periodically could be included into the cost function by a discounting factor assuming a certain project duration and net interest rate. However, these cost elements appear to be more essential if a cost comparison with alternative remediation technologies is envisioned. Here we constrain our study to the use of a "generic" reactive zone that is parameterized only by the cost and hydraulic properties of its filling, the cost of its construction material, and a fixed detention time. The required length of this simplified reactor is, however, still a function of the assumed uniform flow through the gate (see equations (2) and (3) below).

Table 1. Unit Costs for Cost Elements Used for FGS Design Optimization and Parameters Determining the Gate Length

Parameter	Value
Gravel costs C_{Gra}, €/m^3	36
Funnel costs C_F, €/m^2	70
Gate costs C_G, €/m^2	190
t_{det}, hours	72.0
n_e [ ]	0.35

[22] The FGS construction is assumed to be as follows: Funnels perpendicular (funnel widths) and parallel (gate lengths) to the main flow direction are realized as slurry walls. Gates are filled with filter gravel. For stabilization of the gate openings during the filling phase, an up- and a down-gradient sheet pile wall are installed. The unit prices used are based on *Landesumweltamt Nordrhein-Westfahlen* [1998] and listed in Table 1. The total costs (C_T) are calculated by the following formula:

$$C_T = b \cdot \left[C_F \cdot \left(\sum_{i=1}^{n_g+1} w_{f,i} + 2 \cdot \sum_{i=1}^{n_g} l_{g,i} \right) + 2 \cdot C_G \cdot \sum_{i=1}^{n_g} w_{g,i} + C_{Gra} \cdot \sum_{i=1}^{n_g} \left(w_{g,i} \cdot l_{g,i} \right) \right], \tag{1}$$

where b is the aquifer thickness, C_F is the unit price for a funnel segment, C_G is the unit price for the sheet pile segments at the gate openings, and C_{Gra} is the unit price for filter gravel. The appropriate reactor or gate length is determined through the advective flow velocity, $v_{a,i}$, within the gate material and the assumed detention time t_{det},

$$l_{g,i} = v_{a,i} \cdot t_{\mathrm{det}} \tag{2}$$

and likewise can be expressed as

$$l_{g,i} = \frac{Q_i \cdot t_{\mathrm{det}}}{w_{g,i} \cdot b \cdot n_e}, \tag{3}$$

where Q_i is the volumetric flow rate through gate i, n_e is the effective porosity of the gravel material, and $w_{g,i}$ is the width of gate i. The assumed parameter values for l_g determination are presented in Table 1.

2.2. Setup of Numerical Model

[23] The general setup for the performed optimization runs is depicted in Figure 3. A rectangular contaminated area is outlined by particles. By setting additional bounds on decision variables, an area for the FGS emplacement can be defined. The designs suggested by the optimization algorithm are then implemented into the flow model automatically.

[24] All model runs are carried out using the U.S. Geological Survey (USGS) groundwater flow model

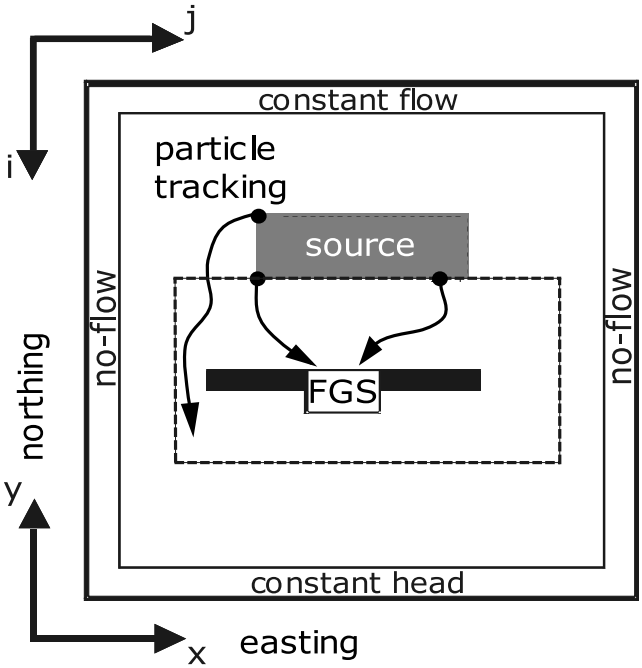

Figure 3. Illustration of the model setup with boundary conditions and coordinate directions. The dashed line delineates an emplacement area for the FGS. The contaminated area is shaded gray.

MODFLOW-2000 [*Harbaugh et al.*, 2000]. The study exclusively covers two-dimensional flow in confined aquifers. This is done mainly to save computation time. It is clear that an unconfined aquifer would be more realistic. However, preliminary testing of a confined versus an unconfined two-dimensional, homogeneous aquifer with a four-gate FGS revealed no tremendous differences in the flow fields. The error introduced is therefore believed to have a negligible effect on the OF in the context of this study. Contaminant transport is simulated by forward particle-tracking using the USGS particle-tracking code MODPATH 4.2 [*Pollock*, 1994]. The volumetric flow rate through individual gates is determined by ZONEBUDGET 2.1 [*Harbaugh*, 1990]. Funnel segments are modeled as horizontal flow barriers with a low conductivity of 1.0×10^{-9} m/s. The gravel filling of the gates is modeled by assigning a conductivity of 0.01 m/s to the corresponding grid cells. The generation of heterogeneous hydraulic conductivity fields was carried out using the GSLIB software package sgsim [*Deutsch and Journel*, 1992]. Each single conductivity realization serves as a heterogeneous conductivity field of a hypothetical aquifer and is assumed to be known with certainty. The underlying random function model assumes a variance of $\sigma^2_{\ln K} = 1.0$ for $Y = \ln K_f$ with a geometric mean of $K_{f,gm} = 1.0 \times 10^{-4}$ m/s (representative for a sand aquifer) and an exponential variogram model with correlation lengths $\lambda_x = 6$ m and $\lambda_y = 12$ m. These properties were taken from ranges of probable characteristics of moderately heterogeneous, sandy aquifers found in the literature [e.g., *Dagan*, 1989; *Moreno and Tsang*, 1994; *Rubin*, 2003]. For the ambient gradient, a general flow direction from north to south is ensured by a northern constant flow boundary and a southern constant head boundary (see Figure 3 for illustration).

[25] The two-dimensional (2-D) flow model used to develop the test case consisted of a 200 × 200 cell grid with a unit cell of 1.0 m × 1.0 m. For the contamination source, a square was used with a side length of 30 m. The *y*-position of the FGS was fixed, so that the latter was placed down-gradient of the source zone by twice the width of the source zone.

[26] For the subsequent, more extensive studies, a 2-D model consisting of a 150 × 300 cell grid with a unit cell size of 2.0 m × 1.0 m was used. The contamination source is represented by a rectangle with a length of 50 m (easting) and a width of 10 m (northing). Because of the uniform grid discretization in both model domains, all funnel and gate widths, as well as gate lengths, are given as (integer) numbers of grid cells. Figure 4 provides a visual impression of the modeled flow domain with potential lines, particles paths, the conductivity field, and a funnel-and-gate design.

2.3. Constraint Formulation

[27] The cost formula (equation (1)) could, at least in principle, be minimized by simply decreasing all widths and lengths to zero. As this contradicts the remediation goals (i.e., plume capture and detention time achievement), the optimization goal "minimize total costs" is also subject to constraints:

$$\min C_T$$

subject to

constraint 1 : full plume capture
constraint 2 : achieve detention time.

[28] In order to examine the suitability of FGS variants to capture the plume in the model, a potential contamination source is outlined by particles. The particles' path lines are then determined and used to discern whether a particle would be captured by gates or not. FGS designs that do not capture all particles are termed invalid, whereas all valid designs capture the total amount of particles starting from the contamination source outline. This approach adopts the advective control strategy for the P&T technology of *Mulligan and Ahlfeld* [1999], which was regarded to be a very general approach, as it does not set any preconditions for the flow field or the optimal control solution [*Bayer and Finkel*, 2004].

[29] **Constraint 1**. Mathematically, the plume capture constraint can be directly incorporated into the OF as a multiplicative penalty term, which depends on the ratio of missed particles and the total number of particles. This ratio serves as a measure for the distance of an invalid solution to the valid solution space. The use of a distance measure is a typical ingredient for penalty term formulations [e.g., *Smith and Coit*, 1997]. In the following we use a formulation similar to that of *Bayer and Finkel* [2004] with a multiplicative exponential penalty term:

$$\Phi(n_{cap}, n_{part}) = p^{q\left(\frac{n_{part}}{n_{cap}}-1\right)^r}, \tag{4}$$

with $n_{cap} \in]0, n_{part}]$ being the number of captured particles out of a total of n_{part} particles. $\Phi(0, n_{part})$ is defined as "infinity." The parameters $p, q, r \in, p > 1, q > 0$, and $0 < r < 1$

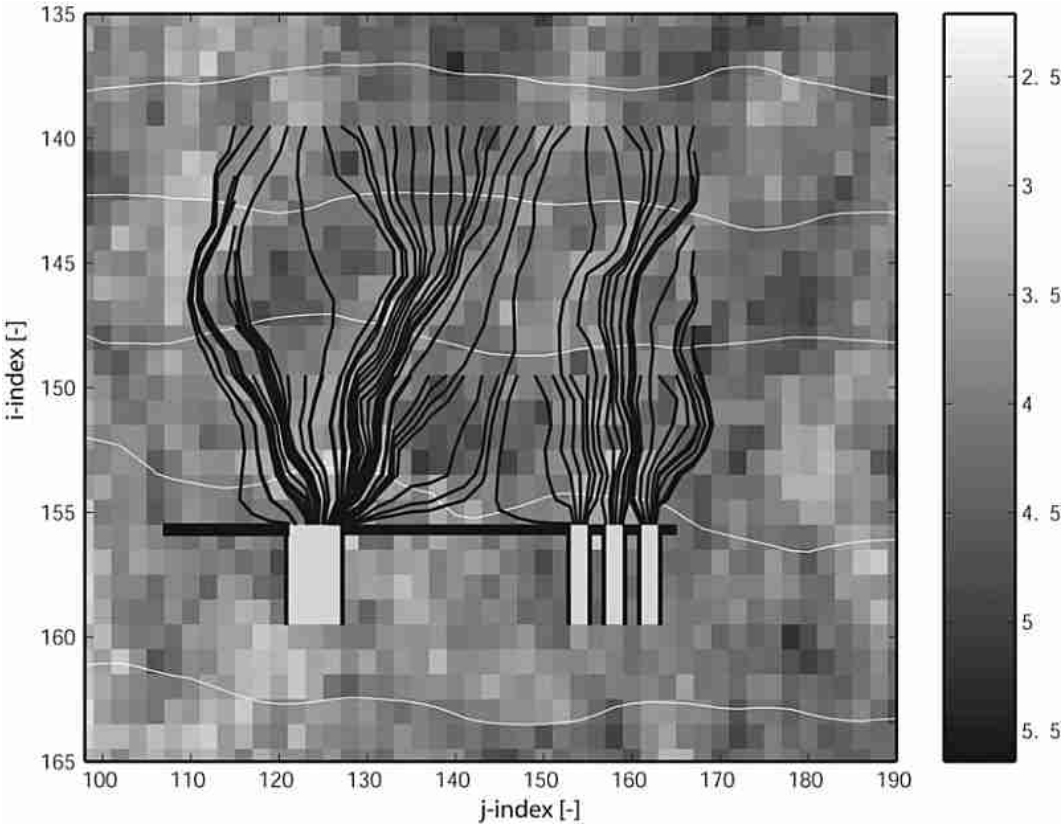

Figure 4. Illustration of an (optimal) FGS design in the template aquifer. The conductivity field is shown in gray scale ($-\log_{10}$ (K_f [m/s])). Particle paths are shown in black and equipotential lines of hydraulic head are shown in white. Please note the different scaling of x- and y-axes.

are user-defined. These parameters were adapted in initial trial runs by ensuring a penalty high enough that invalid solutions are no longer found to be cost-optimal. The above penalty term differs from the formulation used by *Bayer and Finkel* [2004] by using n_{part} over n_{cap} rather than vice versa. The formulation of equation (4) was led primarily by the idea of creating a steep gradient toward the valid solution space for very distant invalid solutions and a more smooth transition closer to the invalid/valid boundary (compare with properties of f(x) = (1/x) − 1 for 0 < x < 1). The penalty term parameter r results in a negative curvature of $\Phi(n_{cap}, n_{part})$ close to the invalid/valid boundary and allows for a step-like rise of the penalization for the ''last missed particle.'' As both evolutionary algorithms only process information based on fitness value ranking, rather than the exact values (or their differences), the algorithms' sensitivity to the particular penalty term formulation is believed to be relatively small if the penalty creates a fitness landscape that is indeed increasing overall, with increasing distance from the valid solution space. The findings reported by *Bayer and Finkel* [2004] for P&T optimization support this conclusion. However, we mention here the caveat that the number of captured particles is not linearly related to the true distance from the valid solution space, i.e., nonlinearity is introduced by a penalty term that is based on the amount of captured or missed particles as a distance measure.

[30] The behavior of $\Phi(n_{cap}, n_{part})$ for $n_{cap} \rightarrow 0$ requires choosing the initial search space reasonably, so that the search does not retrieve a large number of points with ''infinity'' as their OF value. Yet this does not conflict with the aims of the FGS designer, as one tries to initially restrict the search space to exclude extremely unpromising regions.

[31] **Constraint 2**. The achievement of the required detention time is incorporated into the OF by computing a prognosticated gate length based on the given throughputs, Q_i. The latter are obtained from the model runs using a constant gate length (in flow direction) of four model cells. Alternatively, the gate length could be made a decision variable in the optimization process. This is because in a heterogeneous aquifer, the gate length may indeed influence Q_i, for example, through linkage of high-conductivity zones. The constraint of sufficient detention time would then have to be handled independently from plume capture, possibly by a further penalty term. As a compromise, to keep the problem dimensionality low, the required gate length is, in the presented formulation, only driven by Q_i, which, in turn, is a function of gate and funnel widths and is barely influenced by the gate length itself.

[32] Ultimately, the OF (compare to the formulation in the first paragraph of this section) of the optimization problem is reformulated (without gate lengths as decision variables) as follows:

$$\min \mathrm{OF}$$
$$\text{with } \mathrm{OF} = \Phi \cdot C_T\left(x_{pos}, y_{pos}, w_{g,1}, \ldots w_{g,n_g}, w_{f,1}, \ldots w_{f,n_g+1}\right), \tag{5}$$

where upper and/or lower bounds may also be given for each decision variable. Being independent of a model run outcome, such constraints are incorporated a priori by limiting the search space for each decision variable, e.g., $x_{pos} \in [x_{pos,\min}, x_{pos,\max}]$.

2.4. Optimization Algorithms

2.4.1. Simple GA

[33] A simple genetic algorithm encodes decision variable values as 0-1 binary strings or chromosomes. As we are dealing with integer numbers in the FGS design problem stated above, no precision loss arises from the binary encoding and the application of a (simple GA) seems favorable. Following, we provide a short outline of the GA variant used.

[34] The GA uses a population of fixed size (*pop*) of binary string individuals, which is initialized randomly. An optimal solution is determined by a generation-wise application of the successive operations (1) selection, (2) crossover, and (3) mutation. For operation 1 we use tournament selection, where only the fittest individual from a randomly selected group of strings from the current population are allowed to enter the mating pool. According to the recommendation by *Lobo* [2000], a tournament size of 4 was chosen. Among the members in the mating pool the crossover operation performs a coupling of individuals with fixed crossover probability p_c. In the presented study, uniform crossover with p_c set to 0.5 is used, as recommended by *Bäck et al.* [2000]. Hereby, all corresponding bits of two randomly selected mating pool members are swapped with probability p_c to produce offspring individuals (*pop* in total). Finally, mutation is performed on the offspring by flipping the individual bits with a fixed probability of $p_m = 1/pop$. Bounds on decision variable ranges are ensured by discarding invalid binary strings and repeating the generation process. Regarding the population size, we apply the strategy of *Reed et al.* [2000], who suggest that the lower bound on the starting population size should be approximately $1.4L$, where L is the string length of the binary representation of the search space. Depending on the success of the previous population size, *pop* is further increased until the performance is no longer significantly improved.

2.4.2. CMA-ES

[35] The CMA-ES operates on continuous decision variables and differs from typical evolution strategies because of the adaptation of a covariance matrix. A short but mathematically precise formulation of the algorithm is given in Appendix A. For an excellent introduction to (nearly) all algorithmic details, the reader is referred to *Hansen* [1998], *Hansen and Ostermeier* [2001], and *Hansen et al.* [2003].

[36] In the subsequent description, the parameter N expresses the problem dimension, i.e., the number of decision variables to optimize. The (μ_w, λ)-CMA-ES uses information from a population of λ individuals (evaluation points) and chooses the μ_w fittest individuals for property heritage. All λ individuals are drawn randomly from an N-dimensional normal distribution, whose initial mean and initial covariance matrix may be defined by the user. The selection via ranking by fitness determines the μ_w fittest individuals. These are recombined by building a weighted average, i.e., a weighted linear combination of the selected μ_w N-dimensional vectors. This recombined vector represents the new mean of the normal distribution for the next offspring generation. An evolution path is represented by the vector sum of a few consecutive weighted mean vectors, which the search process realized over a certain number of generations. The covariance matrix and a global (scalar) step-size are adapted according to the evolution path. The global step-size is used to realize competitive change rates on a time horizon proportional to N or shorter, since the learning rate via adaptation of the covariance matrix is rather slow, especially for small population sizes. The covariance matrix is adapted to adjust individual step-sizes and directions through dilations and rotations of the principal axes of the contours of equal probability density. By reorientation of these principal axes, correlations between coordinate axes can be accounted for within the search. This means that the algorithm reliably adapts to any general linear transformation (dilation and/or rotation) of the search space. It therefore works independent of the given coordinate system. This makes the CMA-ES eligible for non-separable problems. The adaptation of the global step-size is carried out by comparing the vector sum of the realized steps over a few generations (cumulation of selection information), with the vector sum of the expected steps under random selection. Step sizes are enlarged if the realized evolution path is larger than the expected one, and reduced otherwise.

[37] The handling of a discrete step-function is incorporated by setting a CMA-ES minimum (individual) step size σ_i for each decision variable. According to the recommendation by *Hansen* [1998], it is set to

$$\sigma_i \geq \frac{m_i}{\sqrt{N_{\text{int}}}}, \tag{6}$$

where m_i is the discretization in coordinate i and N_{int} is the number of problem dimensions that have to be limited (within this study, $N = N_{\text{int}}$).

3. Test Case

[38] In order to gain insight into the local topology of the OF, a hypothetical test case with a single gate FGS was created. With a fixed y-position, this simple FGS parameterization already requires a four-dimensional decision space (position in x-direction (easting in Figure 4), one gate width and two funnel widths). The advantage of exploring such a test case is that the optimum can be found with certainty by complete enumeration of all decision variable combinations. This allows one to test the overall capability of the optimization algorithm to identify the optimum. Furthermore, the structure of the fitness landscape itself may be studied.

[39] A heterogeneous conductivity realization was used for the conductivity field with geostatistical parameters as defined above. A preliminary optimization run was completed, and the decision variables were systematically changed around this ''optimum'' setting (incrementing and decrementing each decision variable cell-wise up to a maximum of seven cells from the ''optimum'' setting). The full enumeration of all decision variable combinations needed $15^4 = 50{,}625$ model runs to generate the test set. For each decision variable setting, the number of captured

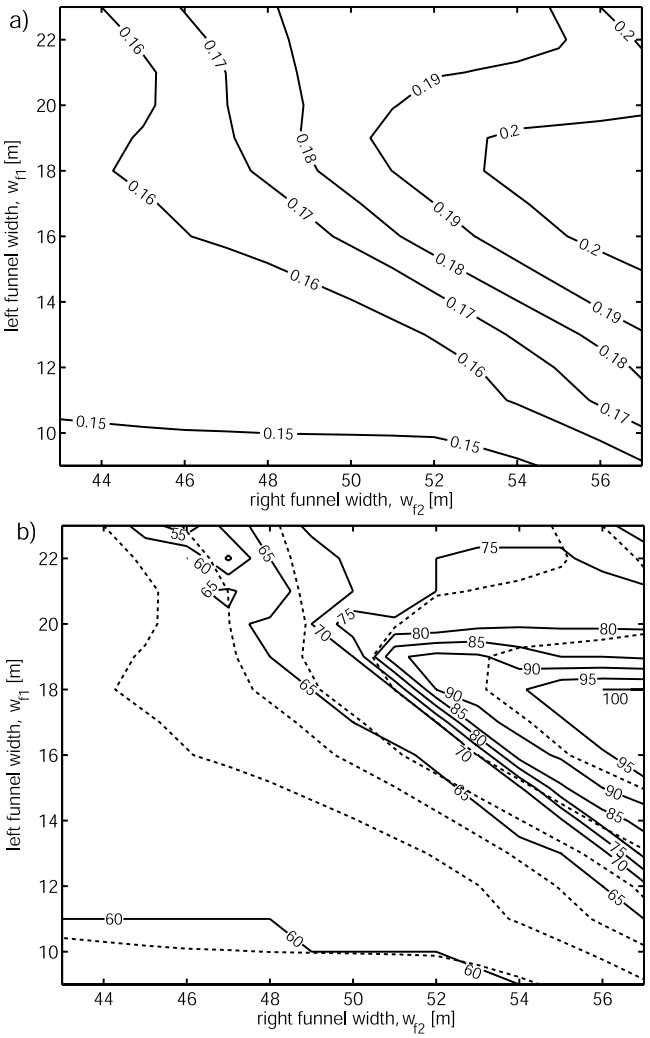

Figure 5. (a) Q distribution of the test case as a function of w_{f1} and w_{f2}. Q is given in L/s. (b) Particle capture distribution (solid n_{cap}/n_{part} contours in percent). Q contours are shown stippled with the values given in Figure 5a. Please note that the 100% capture contour line only comprises two $w_{f1} - w_{f2}$ combinations {(18, 56), (18, 57)} and therefore forms a line rather than a contour.

particles, n_{cap}, and the total volumetric flow rate through the gate, Q, were recorded. A global optimum was found by exhaustive search for the OF (equation (5)) at evaluation point (x_{pos} = 63, w_g = 8, $w_{f,1}$ = 18, $w_{f,2}$ = 56). For the test case, x_{pos} marks the western tip of the FGS, so that, by incrementing x_{pos}, the whole FGS is shifted eastward.

3.1. Results of Test Case

[40] A primary examination of the test case is performed by visualizing the more general functional dependency of the model output parameters (the number of particles captured, n_{cap}, and the flow rate through the gate, Q) on the decision variables. Both model output parameters are incorporated in the OF by equations (1), (3), (4), and (5). How significantly they contribute to the fitness landscape will be dependent on the implicit weighting given in the penalty terms and/or unit prices, respectively.

3.1.1. Cost Surface and the Invalid/Valid Solution Space

[41] In a homogeneous aquifer, one would expect Q to be essentially independent from the FGS x-position (except close to the model boundaries) and follow a roughly linear trend with increasing funnel or gate width. The test case proves that this is not true in a heterogeneous aquifer.

[42] For visualization matters, we restrict the presentation of results to the functional relationship of funnel widths $w_{f,1}$ and $w_{f,2}$ (with x_{pos} and w_g set to the global optimum values) and the respective output parameters (Q, n_{cap}, C_T, fitness). These relationships appear to depict the most important features sufficiently. Figure 5a shows the Q distribution within the test set as a function of $w_{f,1}$ and $w_{f,2}$. The rather explicit nonlinearity stems from the heterogeneous conductivity field, which inhibits or promotes flow through the gate according to the gate's position. Please note that by increasing $w_{f,1}$, the gate position is also shifted, whereas $w_{f,2}$ can be changed without influencing the gate position.

[43] The contours delineating the percentage of captured particles (n_{cap}/n_{part}) in the $w_{f,1}-w_{f,2}$ plane are shown in Figure 6b (lain over the stippled Q contours of Figure 5a).

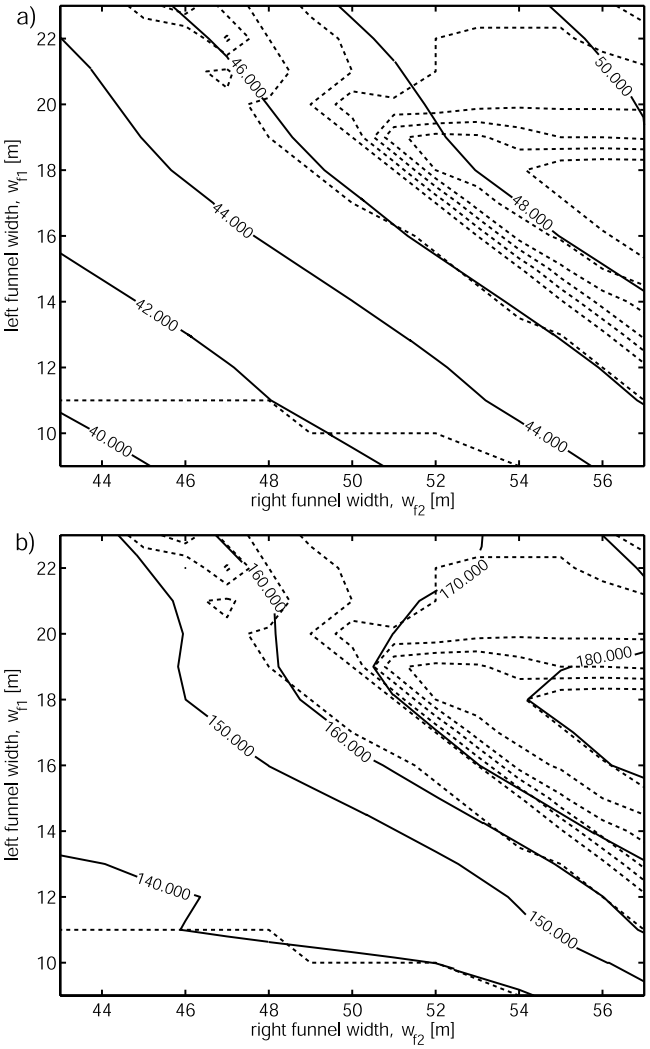

Figure 6. Cost surface as a function of w_{f1} and w_{f2} for the (a) inexpensive and (b) expensive gate reactor with particle distribution (stippled).

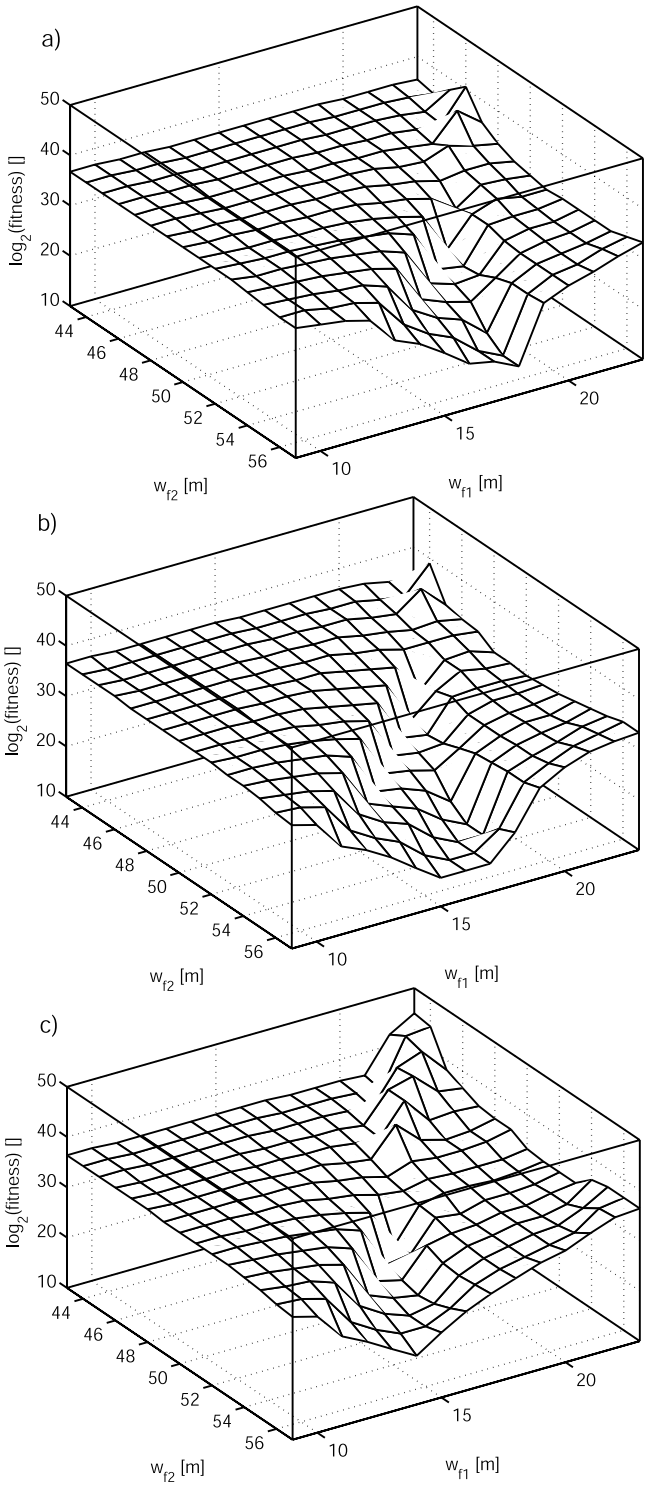

Figure 7. (a) Fitness landscape around the global optimum of the test case. (b) Fitness landscape with w_g incremented by two cells. (c) Fitness landscape with x_{pos} incremented by two cells.

Even though far from matching, both contours seem to follow the same general trend. Another remarkable feature is that, for example, for a fixed $w_{f,1} = 14$, n_{cap}/n_{part} is above 75 for $w_{f,2} = 12$, but n_{cap}/n_{part} is below 75 if $w_{f,2}$ is incremented from 12 to 14. This means that even though neither width ($w_{f,1}$, w_g) nor position of the first funnel (x_{pos}) or the gate ($x_{pos} + w_{f,1} + 1$) are changed, an overly large second funnel may also worsen the overall performance. Visual inspection of such cases found that the overly large second funnel raised the ratio of Q coming from the eastern part of the aquifer, and caused a decrease in particle capture on the side of the western funnel. This finding may be considered a clue to the hypothesis that correlation between decision variables is inherent to the problem.

[44] The final fitness landscape is shaped by the transformation through the cost computation. The OF would be essentially linear if the gate length l_g was constant. Since l_g is calculated as a function of Q, nonlinearity enters the cost function through the Q distribution (Figure 5a). Whether or not this nonlinearity becomes a dominant feature depends on the assigned weighting of l_g through the different unit costs. Figure 6a shows the computed costs (solid contours) along with stippled n_{cap}/n_{part} contours. In the cost function predominantly used in this study, the gate reactor is a rather inexpensive one (see Table 1). Most of the irregularity due to the Q distribution is smoothed out, as the cost contours seem to follow a roughly linear trend. It can also be inferred that the n_{cap}/n_{part} distribution will cause most of the "roughness" of the fitness landscape through the penalty term. Yet we can expect that more expensive reactor types exist. For illustration purposes (see Figure 6b), the price of the reactor filling was raised to 900 €/m^3, which is an approximate unit cost for a zero-valent iron reactor [e.g., *Bürger et al.*, 2003a]. Here the cost contours seem to mimic those of the Q distribution (see Figure 5a), creating a more complex cost surface.

3.1.2. Fitness Landscape

[45] In order to examine the variability of the final fitness landscape (i.e., the penalized cost function) more closely, Figure 7 shows a three-dimensional plot of the fitness function of the test case (for reference cost parameters, see Table 1). Please note that the z-axis is $\log_2$-transformed, so that the deceptive "plateau" does, in fact, exhibit several undulations (local minima). The most prominent feature is a rugged ridge at an oblique angle to the coordinate axes, which is transforming into a valley, where the optimum is located. The task for the optimization algorithms is therefore to locate this valley and to follow it along its axis (obliquely to the coordinate axes). Unfortunately, this valley not only is dependent on $w_{f,1}$ and $w_{f,2}$, but also extends into the two other dimensions, x_{pos} and w_g, which were previously held constant. A variation in x_{pos} is expected to shift an otherwise feasible FGS design into unfeasible regions and vice versa. A variation in w_g shifts the right funnel and is also expected to increase the volumetric flow rate through the gate. Therefore it should raise costs but should also influence the penalty. In order to gain a visual impression of these changes, Figure 8b and Figure 8c are depicted, in which both, x_{pos} and w_g, are incremented by two cells.

[46] An increase in gate width (Figure 7b) seems to widen the valley, but at the same time the valley is shifted toward a lower $w_{f,1}$. From the actual numbers in the data set, it is also evident that a wider gate shifts the cost surface upward. A change in position, as expected, shifts the valley, but the valley itself also transforms, as can be seen in Figure 7c (due to the irregular n_{cap}/n_{part} and Q distributions).

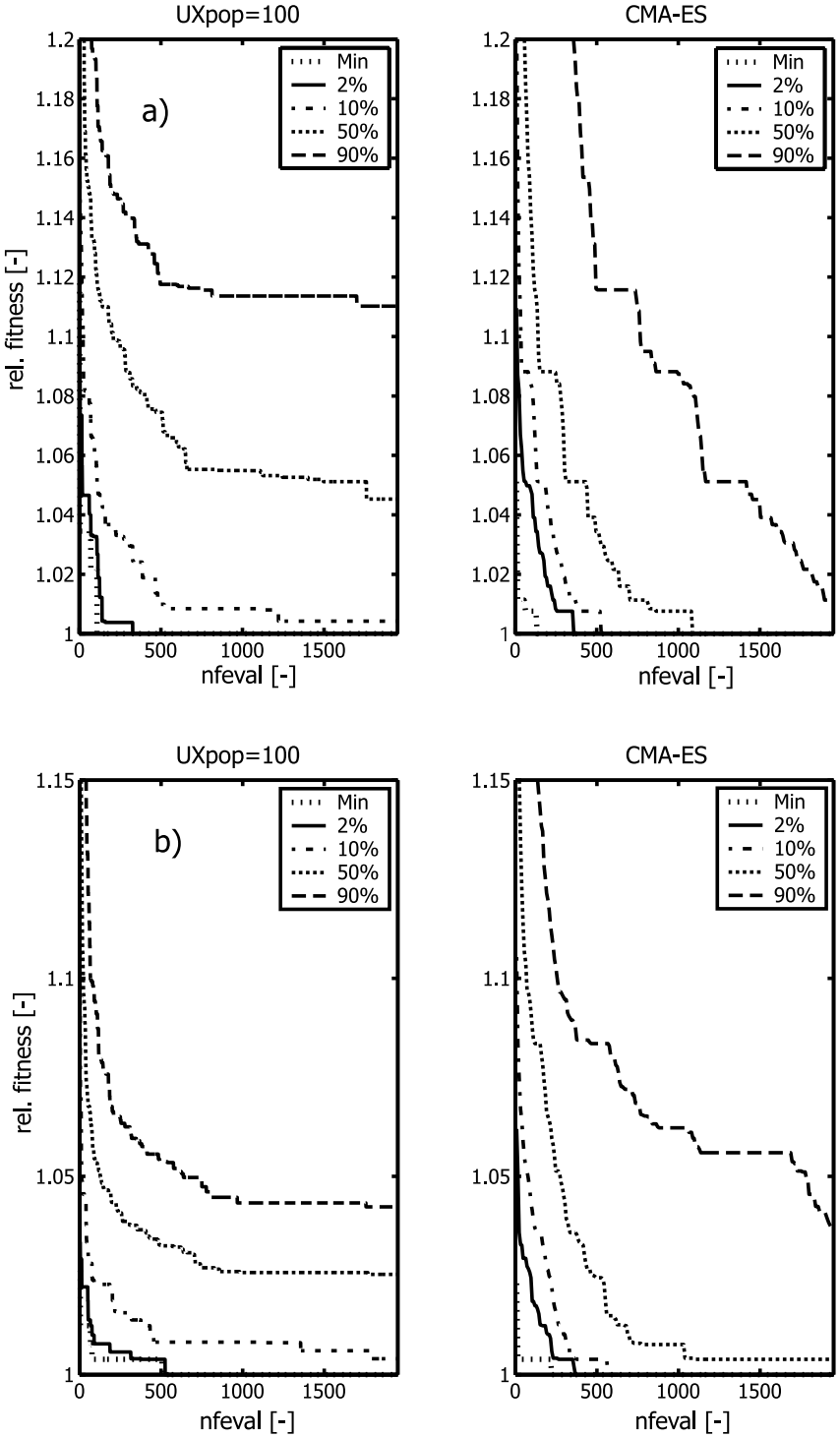

Figure 8. Cumulative frequency distributions for the simple GA (U × *pop* = 100) and the CMA-ES for the test case: (a) inexpensive gate reactor and (b) expensive gate reactor.

[47] To sum up the findings from the test case analysis, the OF used exhibits correlations in all four object parameters (in this encoding) and does have some local minima. In terms of the existence of local gradients, the fitness landscape might be attributed to an occasional ruggedness (ridge), but the overall appearance seems to exhibit a notion of continuity rather than randomly arranged fitness peaks and/or holes. This is taken as evidence of the validity of the strong causality assumption.

3.2. Optimization Runs for the Test Case

[48] One hundred independent optimization runs (ORs) where performed (n_{OR} = 100) to optimize the one-gate FGS of the test case with the simple GA of varying population size and the CMA-ES. The starting population size for the GA was determined as $1.4L \approx 25$ (with L = 16 = four

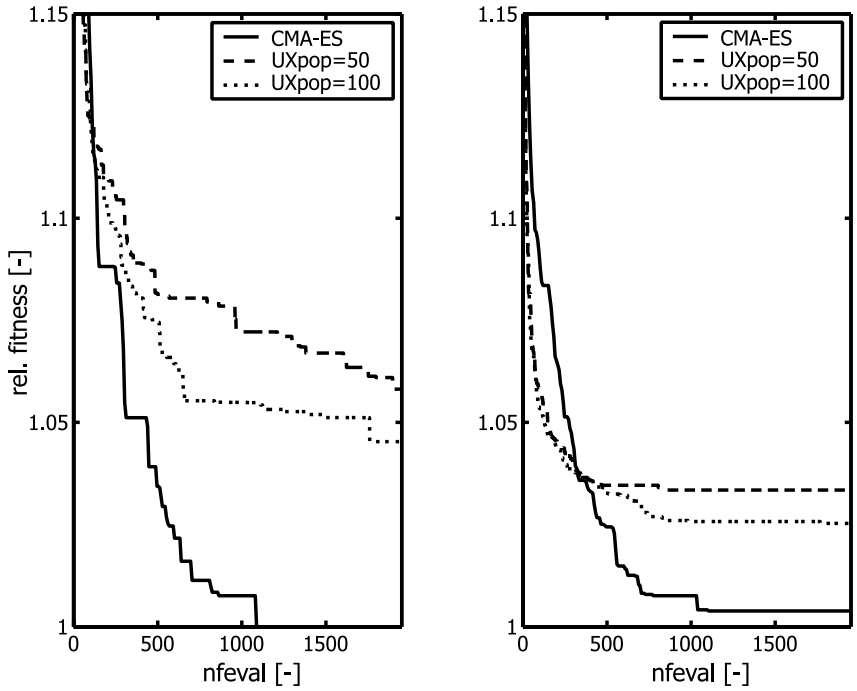

Figure 9. Comparison of median optimization runs for simple GAs with varying population sizes and the CMA-ES: (left) inexpensive gate reactor and (right) expensive gate reactor.

decision variables times four encoding bits per decision variable). The starting points for CMA-ES runs where chosen randomly throughout the search space. The maximum number of OF evaluations (i.e., groundwater model calls) was set to 2000.

[49] The results of these ORs are presented in Figures 8 and 9 Because of the stochastic nature of both algorithms, the obtained best fitness values for a certain *nfeval* may be interpreted statistically over the number of algorithm restarts (n_{OR}). Following the approach of *Bayer and Finkel* [2004, 2006b], cumulative frequency distributions are constructed, which assign a probability P as a function of *nfeval* to a certain fitness, f_0, according to the frequency of obtaining this value (or a lower one) over all n_{OR}:

$$P = \text{Prob}(\text{OF}(nfeval) \leq f_0). \tag{7}$$

[50] Algorithm performance may also be measured by the smallest fitness value reached on average for a specified *nfeval*. In this study we use percentiles rather than moments (e.g., median rather than mean) for distribution definition, in order to be less dependent on the individual fitness values. Furthermore, we normalize all fitness values by the global optimum.

[51] Figure 8a depicts the algorithms' performance for the test case. For the GA the best results obtained with *pop* = 100 are shown (*pop*s up to 200 did not result in significant improvements). Both algorithms succeed in locating the global optimum within less than 500 *nfeval* up to the 2nd percentile of all ORs. Furthermore, 90% of all ORs resulted in fitness values better than within 12% of the global optimum. However, there is a clear difference regarding the robustness and success probability between the applied GA and the CMA-ES. The GA percentile curves level off at around 700 *nfeval* and show little progress toward the global optimum. In case of the CMA-ES, on the other hand, the performance distribution shows a clear trend toward the global optimum. At maximum *nfeval*, 70% of all ORs have successfully located the global optimum of the test case.

[52] As stated above, the contribution of the volumetric flow rate through the gate to the fitness landscape undergoes weighting by the reactor costs. Figure 8b shows the results for a second optimization study regarding the GA and CMA-ES on the test set with the increased reactor costs. The results of these ORs show that a higher weighting of the (heterogeneously distributed) volumetric flow rate seems to create a more complex problem, as the number of identified global optima decreased (GA, 10 versus 6; CMA-ES, 70 versus 43). Furthermore, the distribution of the CMA-ES also shows a leveling off in the median and the 90th percentile. Yet in comparison with the GA, the CMA-ES generally seems to more robustly locate the global optimum, and with a smaller *nfeval* in the best OR.

[53] In a more direct comparison, Figure 9 shows the median performance of the GA (*pop* = 50, 100) and the CMA-ES for the inexpensive and the expensive reactor cases. The median performance should be the most reliable, since the stochastic nature of both algorithms could always allow for a particularly bad or good OR. As can be seen in Figure 9, the GA may, indeed, locate relatively good fitness values, very efficiently, within only a few *nfeval* (expensive reactor case). However, compared with the CMA-ES it seems to stagnate at considerably less optimal fitness values. The larger population size does improve the GA performance; however, the leveling-off appears to be characteristic. In view of these findings, we focus exclusively on the application of the CMA-ES in the second part of this

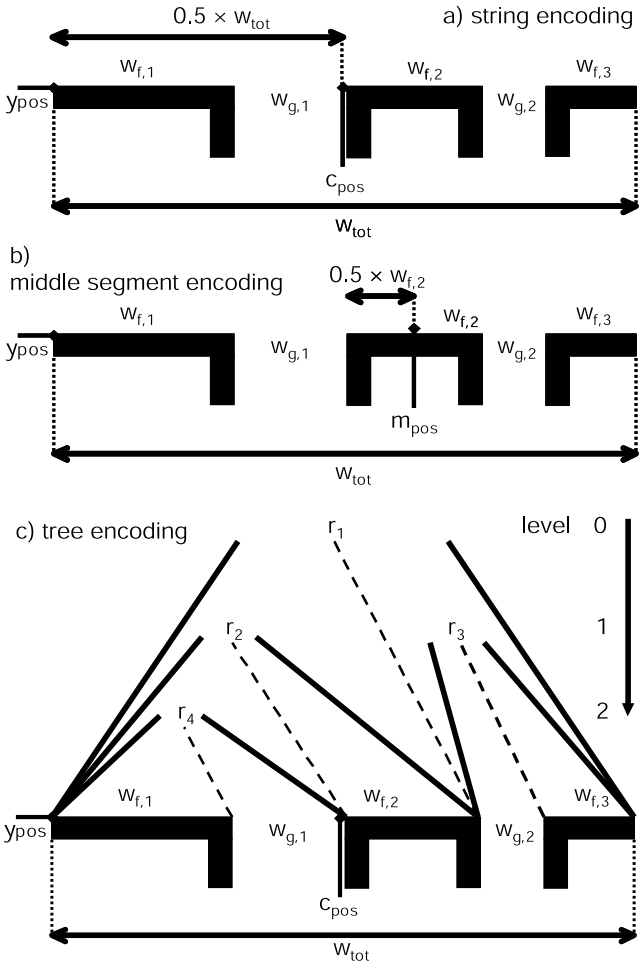

Figure 10. Illustration of (a) string, (b) middle segment (MSE), and (c) tree encoding for $n_g = 2$. For tree encoding, individual segment widths ($w_{f,i}$ and $w_{g,i}$) are represented by a hierarchical partitioning of w_{tot} via the ratios r_i.

article, where a more complex FGS design problem is defined.

4. Problem Encoding

[54] In general, problem encoding is a tricky task in optimization. The following section introduces three different types of encodings that were specifically developed for the purpose of algorithmic FGS design optimization. From an optimization perspective, it is favorable to have as few decision variable correlations as possible. In other words, the problem then becomes "more separable." Additionally, it is favorable for the sensitivity of the fitness, regarding a decision variable, to be the same (at least locally) for all decision variables (i.e., no mis-scaled coordinate axes). It is therefore advisable to seek a problem encoding which comes close to these objectives. Subsequently, three different encoding procedures are presented for comparison.

[55] Principally, it would have been useful to let the optimization algorithm choose the ideal number of gates, problem-dependent. In this study the number of gates n_g is not used as a decision variable, but is defined a priori. The parameter n_g determines the problem dimension: $N = 2 \times n_g + 3$ (without considering gate lengths). If n_g were a decision variable, then the search would have to degenerate into a subspace of the original search space in order to find an optimal solution with an n_g lower than an arbitrary maximum specified. Thereby, some decision variables have to be made ineffective, for example, by setting them to exactly zero. It was therefore assumed that it would be more efficient to optimize for each n_g individually. This kind of problem decomposition concurs with findings reported by *Zheng and Wang* [1999] for P&T optimization with varying numbers of wells.

[56] Hereinafter, funnel and gate widths (opposed to gate flow-through lengths) are referred to as FGS segments. Theoretically, optimization parameters could represent FGS segment position coordinates and/or dimensions (widths). The problematic feature of the modeled straight FGSs is that funnel tips represent gate beginnings and vice versa. If more than one segment position is a decision variable, the decision variable values must have the same linear order as the segments they represent. For example, the funnel east of the gate in Figure 3 must not have an assigned x-position that is west of the tip-position of the gate. This implies that positions would have to be chosen from mutually exclusive intervals that maintain the given segment order. Generally, however, this means a drastic predefinition of the obtainable optimal solutions. In practice, such an encoding may only be useful for a single gate FGS ($n_g = 1$).

[57] Using only one position, e.g., the starting position of the westernmost funnel (subsequently referred to as x_{pos} as in the test case), and modeling the FGS design via its segment dimensions is a way to maintain solution flexibility. However, most decision variables will be correlated. For example, if the starting position of the first funnel, x_{pos}, is fixed and its width $w_{f,1}$ is increased by n grid cells, all following adjacent segments will also be shifted in position by n grid cells. This shift in position is undesirable as, in order to simply shrink a reasonably positioned system, nearly all decision variables have to be changed consistently.

[58] A relatively simple remedy is to assign x_{pos} to the center of the FGS (c_{pos}) and to build the FGS around it (as depicted in Figure 10a). With this encoding, at least the overall position of the FGS is independent of other object parameter changes. The starting position (westernmost grid cell representing the segment) of the ith gate segment, $w_{pos}^{g,i}$, can then be calculated by

$$w_{pos}^{g,i} = c_{pos} - \left\lfloor \frac{1}{2} \cdot w_{tot} \right\rfloor + \sum_{k=1}^{i} w_{f,k} + \sum_{j=2}^{i} w_{g,j-1}, \qquad (8)$$

where $i = 1,\ldots,n_g$ and $\lfloor x \rfloor$ denote the floor function (i.e., it returns the greatest integer smaller or equal $x \in \mathbb{R}$). The starting position of the ith funnel segment, $w_{pos}^{f,i}$, is given by

$$w_{pos}^{f,i} = c_{pos} - \left\lfloor \frac{1}{2} \cdot w_{tot} \right\rfloor + \sum_{k=2}^{i} \left(w_{g,k-1} + w_{f,k-1} \right), \qquad (9)$$

with $i = 1,\ldots,n_g + 1$. The total FGS width, w_{tot}, is defined as

$$w_{tot} = w_{f,n_g+1} + \sum_{i=1}^{n_g} \left(w_{g,i} + w_{f,i} \right). \qquad (10)$$

Owing to the fact that all segment positions are defined by concatenation of widths, this type of problem encoding is termed "string encoding." A principle advantage of this encoding is that a minimum step-size can be easily and individually assigned to each decision variable in order to deal with the model discretization. However, apart from the still critical correlations among the decision variables, a practical problem with string encoding is that in order to allow a large number of possible FGS designs, the limiting ranges on individual funnel widths and gate widths have to be relatively large. Hence it may not be possible to restrict FGS locations to a specified (rectangular) area and at the same time maintain maximum solution flexibility. If such further restriction is desired, either an additional penalty term has to be introduced or a different problem encoding has to be used.

[59] As a second variant, the so-called "tree encoding" was developed (Figure 10c). The FGS is parameterized by its center position c_{pos} and its total system width w_{tot}. Apart from the y-position, y_{pos}, all other decision variables represent ratios $r_i \in]0, 1[$, which partition w_{tot} in a hierarchical way (a binary tree). For example, ratio r_1 splits w_{tot} in two parts according to $r_1 \times w_{tot}$ and $(1 - r_1) \times w_{tot}$, which are then individually subdivided by r_2 or r_3, respectively. The height of the tree h_{tr} or number of tree levels is given by the number of FGS segments, which are ultimately a function of $n_g > 0$.

$$h_{tr} = 1 + \lfloor \log_2 (n_g) \rfloor, \tag{11}$$

with $\lfloor \cdot \rfloor$ denoting the floor function (as defined above).

[60] As visualized in Figure 10c, an adjustment of the parameters r_4 and/or r_3 does not have an influence on the position of $w_{f,2}$, nor does an adjustment of one of these ratios influence the design settings of the other parameters. Even though this problem encoding is (owing to the hierarchical tree structure) still far from free of decision variable correlations, it provides an increasing number of independently adjustable parameters as n_g increases. It is therefore expected to be an advantageous encoding, especially for multigate FGS.

[61] Tree encoding also allows the replacement of the center position c_{pos} and total FGS width w_{tot} by a starting position x_{st} and an ending position x_{end} (the total width, w_{tot}, is given by $x_{end} - x_{st} + 1$). Then, both positions could be chosen from two mutually exclusive intervals, $[x_{st,\min}, x_{st,\max}]$ and $[x_{end,\min}, x_{end,\max}]$, such that $x_{end,\min} - x_{st,\max} + 1$ represents the minimum total system width, $w_{tot,\min}$. Hence a space constraint (given by a rectangular area within which the FGS has to be placed) could be easily enforced by adjusting $x_{st,\min}$ and $x_{end,\max}$, accordingly. However, this tree encoding variant reduces the solution flexibility, as the interval from $x_{st,\max}$ to $x_{end,\min}$ will be a part of every FGS design that could be suggested by an optimization algorithm. Thus, and for the sake of better comparability with the other encodings, this latter variant was discarded and only the encoding via c_{pos} and w_{tot} was used in this study.

[62] A principal hindrance for tree encoding stems from the discrete nature of the flow model realization of the individual widths. Theoretically, a minimum increment for the ratios r_i has to be set to avoid evaluation points being set too closely, as these would all obtain the same fitness. For example, if a hundred grid cells were to be split into two parts, only ratios differing in at least 1/100 could ensure different splits. The problem that arises becomes clear when looking at the next tree level. If these hundred cells were split according to $r = 30/100$, the minimum increment for ratios on the next tree level would be 1/30 on the left branch and 1/70 on the right branch. Because of the tree structure, the minimum increment for r_i on higher tree levels is dependent on the actual realized partition on the lower tree levels. This means that no a priori limits can be given. It follows that tree encoding could lead to problems when used with the CMA-ES on a discrete problem. Yet, because of the stochastic nature of the optimization algorithm and the fact that a whole population of individuals is used for selection, the minimum increment problem does not limit the practical application of this method. An error only occurs if all fitness values in one population are the same. At least in the ORs performed for this study, a large enough diversity could be ensured within the populations by choosing 0.01 as the minimum ratio increment. If individual segments could be represented by the flow model as real values (e.g., by adaptive grid refinement), this disadvantage of tree encoding would be eliminated immediately.

[63] The third method of encoding developed for this study is "middle segment encoding" (MSE; see Figure 5b). It is almost identical to string encoding, but rather than x_{pos} representing the center of the FGS, it is assigned to the center of the middle segment (m_{pos}). The middle segment is a gate if n_g is odd, and a funnel if n_g is even. In contrast to string encoding, MSE allows the two outermost funnel widths to be adjusted without shifting any other segments. The advantage of decoupling the overall position of the FGS from individual decision variable changes is only partly lost, since the middle segment is somewhat expected to be representative for the center of the FGS in most design settings. Apart from that, this encoding suffers from the same practical problems as string encoding, but shares the advantage of a priori specifiable minimum step-sizes.

[64] For simplicity, let n_g be even. Then the index of the middle funnel, id_{mf}, is given as $\lceil \frac{1}{2}(n_g + 1) \rceil$, with $\lceil x \rceil$ denoting the ceiling function (i.e., it returns the smallest integer greater or equal $x \in \mathbb{R}$). The starting position of the ith ($i = 1,\ldots,n_g$) gate segment, $w_{pos}^{g,i}$, is defined by

$$w_{pos}^{g,i} = m_{pos} - \left\lfloor \frac{1}{2} \cdot w_{f,id_{mf}} \right\rfloor - \sum_{k=1}^{id_{mf}-1} \left(w_{f,k} + w_{g,k} \right) + \sum_{k=1}^{i} w_{f,k} + \sum_{j=2}^{i} w_{g,j-1} \tag{12}$$

and similarly the starting position of the ith ($i = 1,\ldots,n_g + 1$) funnel, $w_{pos}^{f,i}$:

$$w_{pos}^{f,i} = m_{pos} - \left\lfloor \frac{1}{2} \cdot w_{f,id_{mf}} \right\rfloor - \sum_{k=1}^{id_{mf}-1} \left(w_{f,k} + w_{g,k} \right) + \sum_{k=2}^{i} \left(w_{f,k} + w_{g,k} \right). \tag{13}$$

4.1. Comparison of Three Different Problem Encodings

[65] The inspection of the test case revealed initial insights into the complexity of the fitness landscapes and the capabilities of the optimization algorithms. However,

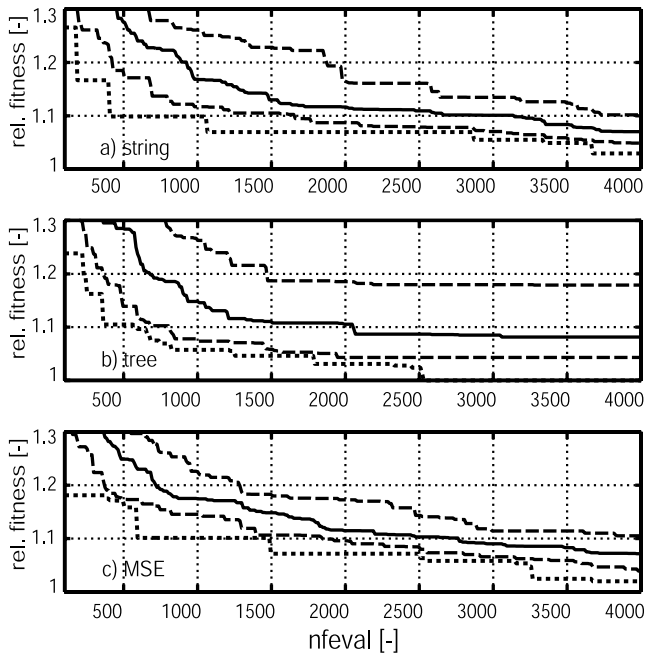

Figure 11. Cumulative frequency distributions for the Gauss1 case: (a) string encoding, (b) tree encoding, and (c) MSE encoding. The minimum fitness is shown as a stippled contour. Dashed contours represent the 10th and 90th percentile, and solid contours represent the median values.

the test case covered a relatively small search space compared with the demands of a more realistic optimization scenario, and only the minimum number of decision variables was used. As previous studies recommend the use of multigate FGS designs to improve hydraulic efficiency [*Starr and Cherry*, 1994; *Bürger et al.*, 2003b; *Bürger and Finkel*, 2005], the optimization procedure was tested on four-gate FGS designs. In addition, the search space was significantly increased. The overall problem complexity was also raised, as a four-gate FGS has a problem dimension of $N = 11$. The resulting search space is no longer searchable by complete enumeration in a reasonable amount of computational time. It follows that the true global optimum is not known with certainty, and only the best solution found can be used as a reference point for near-optimal solutions.

[66] In the following, the speed-up potential of the three problem encodings introduced above is investigated for three different heterogeneous aquifer realizations (Gauss1 to Gauss3, $K_{f,mean} = 10^{-4}$ m/s, $\sigma^2_{\ln Kf} = 1.0$, $\lambda_x = 6$ m, $\lambda_y = 12$ m, rectangular contamination source area as stated above). The highest speed-up effect is expected for a reduction of decision variable correlations, which have to be learned by the CMA-ES as the search proceeds.

[67] It seems logical for an objective comparison, that both the size of the (discrete) search spaces (i.e., the number of different decision variable combinations) and the set of different implementable FGS designs should be the same for each encoding. However, this can not be ensured in this study, due to the peculiarities of the encodings. For instance, using tree encoding, the total system width can be set by a single decision variable, whereas for the other encodings, it is the sum of all segment width parameters. The latter have to be bounded to ensure a maximum total width; however, for tree encoding bounds on the individual ratios cannot be given. Therefore tree encoding will always allow for individual funnel widths larger than the specified upper bound for the other two encodings (unless the total system width is set to that bound; this, however, makes no sense).

[68] The bound constraints given below represent a compromise which follows the reasoning that at least the designs of a larger vicinity of the expected optimal region should be equally explorable or implementable by the different encodings. This is enforced through bounds on tree encoding that generate a search space that is larger than those of string encoding and MSE. However, only unusually large designs, which are not expected to belong to the region of optimal solutions, make up for the extension of the tree encoding search space. It is therefore expected that the overall search progress is not dramatically influenced by the lower solution flexibility of string and MSE encoding for the regions of unusually large designs.The bound constraints on the individual decision variables are the following:

String encoding

$$\left(c_{pos} \in [45, 100], y_{pos} \in [151, 185], w_{g,i} \in [1, 5], w_{f,i} \in [1, 30]\right)$$

Tree encoding

$$\left(c_{pos} \in [45, 100], y_{pos} \in [151, 185], w_{tot} \in [9, 170], r_i \in \left]0, 1\right[\right)$$

MSE

$$\left(m_{pos} \in [45, 100], y_{pos} \in [151, 185], w_{g,i} \in [1, 5], w_{f,i} \in [1, 30]\right).$$

[69] In addition, other intricacies surface for an objective comparison. Theoretically, one could use the same starting point (i.e., a decision variable combination representing the same FGS design) for all encodings. Then several optimization runs are evaluated in terms of the obtained minimum fitness and with respect to the number of times this value was found. Quite naturally, the same FGS design is represented by different points within the search space for the different encodings. This also means, though, that the initial covariance matrix cannot be chosen in such a way that the initial distribution covers the same range of FGS designs. It was therefore considered to be better to use randomly selected starting points.

[70] The actual optimization runs were carried out within an 11-dimensional search cube of edge length 1 ($[0, 1]^{11}$), into which the original search spaces were linearly transformed. This makes "half" the unity matrix, $\mathbf{I}$ (of size 11×11), the most reasonable choice for the initial covariance matrix (set to $0.5 \times \mathbf{I}$) (N. Hansen, personal communication, 2004). The starting point (mean) of the initial distribution of the CMA-ES was decision variable-wise randomly selected from the [0, 1]-interval. The maximum *nfeval* was set to 4000 for each run and the number of optimization runs, n_{OR}, was set to 30.

[71] Cumulative frequency distributions, as defined for the test case, are also used for the comparison of the encodings. For realization Gauss1, they are shown in Figure 11. The best solution was identified with tree

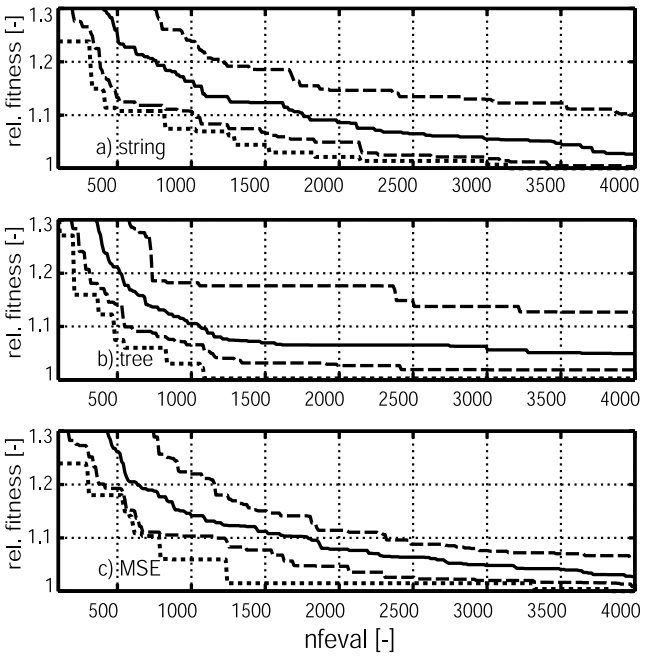

Figure 12. Cumulative frequency distributions for the Gauss2 case: (a) string encoding, (b) tree encoding, and (c) MSE encoding. The minimum fitness is shown as a stippled contour. Dashed contours represent the 10th and 90th percentile, and solid contours represent the median values.

encoding. Up to *nfeval*s of 2000 tree encoding yields smaller median fitness values than both other encodings. It is also better in the 10th percentile up to 3500 *nfeval*. A dominant feature of the tree encoding curves is, however, the loss of progress around 2000 *nfeval*. The curves hardly seem to improve, except for the minimum curve. Quite to the contrary, string encoding and MSE both show continuously improving fitness values in all parameters. The ranges between the 10th and the 90th percentile are also much smaller for string encoding and MSE than for tree encoding.

[72] For realization Gauss2, the best solution was identified with both string encoding and MSE (Figure 12). Tree encoding, again, yields smaller median fitness values than both other encodings up to *nfeval*s of 2500. Similar to Gauss1, it performs better in the 10th percentile up to 3000 *nfeval*. While tree encoding again shows a loss of progress around 1500 *nfeval*, string encoding and MSE yield continuously improving fitness values and eventually reach better values in all percentiles above *nfeval* = 3000. The ranges between the 10th and the 90th percentile are again smaller for string encoding and MSE than for tree encoding. In fact, MSE has the smallest range of all.

[73] For Gauss3 the best solution was most often identified with tree encoding (Figure 13). Compared with the other two cases, it seems that Gauss3 is a rather difficult case for optimization. Tree encoding shows the best performance up to *nfeval*s of 1500 in the median. In the 10th percentile, it is superior up to around 2700 *nfeval*. Again, though, the loss of progress is quite noticeable, and the 90th percentile is far off, compared with the other two encodings. Accordingly, the two other encodings perform better for higher *nfeval*. The ranges between the 10th and the 90th percentile are comparable for both string encoding and MSE.

[74] As a summary of results, one can state that tree encoding provided the fastest improvement in fitness values at the beginning of the search but suffers from a loss of progress as the optimization reaches a certain state of maturity. We interpret these findings as follows. The high level of progress of tree encoding at the beginning of the search shows that the higher number of independently adjustable decision variables does in fact speed up the search process. This is especially the case as tree encoding also covers a larger search space. The loss of progress is most likely related to the problem of a priori minimum step-sizes. The preselected minimum step size of 0.01 is likely to be too small for higher tree levels once fine tuning becomes necessary. At present, there is no real remedy for this problem, unless the model grid could be made adaptive so that real valued segment widths could be represented (minimum step-sizes become unnecessary). A practical approach for future research would be to choose tree-level-dependent minimum step-sizes (e.g., the higher the level, the larger the minimum step-size). This, however, would require preliminary optimization runs, or a good guess of the range of the segment widths within the optimum region. A further grid refinement does not eliminate the theoretical problem, but is expected to postpone the loss of progress to practically irrelevant step-sizes. However, it would also increase the cost of the required computation time for a single model run.

[75] For string encoding and MSE, the courses of the distribution curves suggest that the adaptation of the CMA-ES will further continue for *nfeval* > 4000 for most ORs. On one hand, this shows the slower convergence of

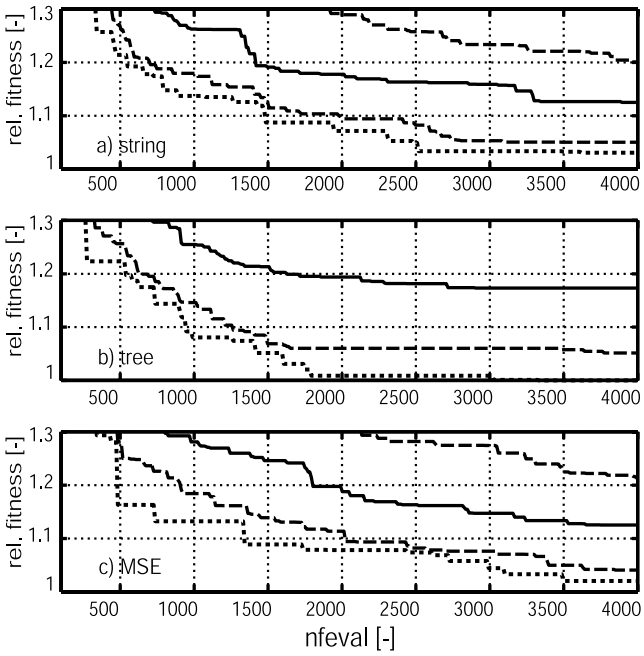

Figure 13. Cumulative frequency distributions for the Gauss3 case: (a) string encoding, (b) tree encoding, and (c) MSE encoding. The minimum fitness is shown as a stippled contour. Dashed contours represent the 10th and 90th percentile, and solid contours represent the median values.

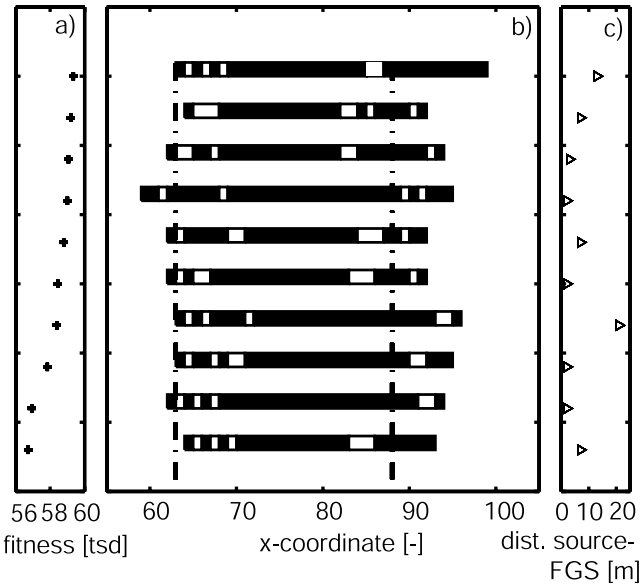

Figure 14. Ten best designs found for Gauss1: (a) fitness and (b) FGS designs and *x*-position. (The *x*-position of the source zone is indicated by the two vertical dashed lines.) Please note that only the *x*-direction is scaled. (c) Distance between FGS and source zone.

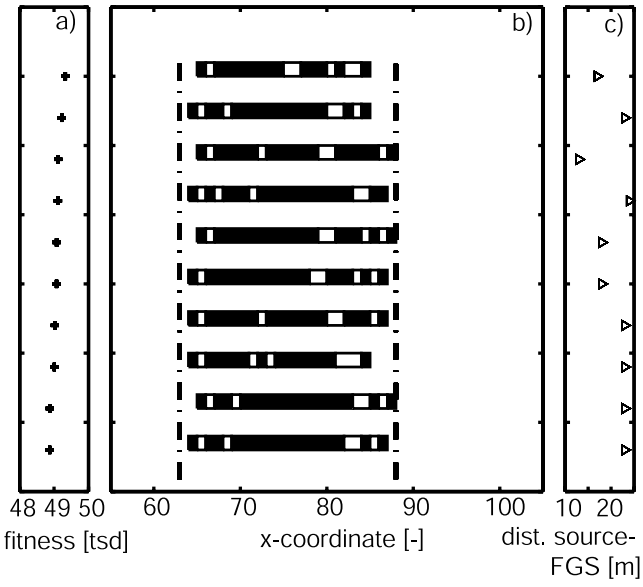

Figure 15. Ten best designs found for Gauss2: (a) fitness and (b) FGS designs and *x*-position (The *x*-position of the source zone is indicated by the two vertical dashed lines.) Please note that only the *x*-direction is scaled. (c) Distance between FGS and source zone.

these encodings compared with tree encoding. On the other hand, they are likely to result in more robust searches, considering the consequent decrease in distribution widths (range between the 10th and 90th percentile).

4.2. Discussion of Obtained Optimal and Near-Optimal Solutions

[76] As mentioned above, a practical interest led to the investigation of near-optimal solutions. For this purpose, the 10 best designs obtained from either of the encoding optimization runs were visually inspected for realizations Gauss1, Gauss2, and Gauss3, and are shown in Figures 14 to 16.

[77] For Gauss1 it is relatively obvious that a large central barrier surrounded by closely spaced gates appears as a general pattern. Even though the distances between the fixed source zone and the FGS designs, as well as the *x*-positions, differ, the overall designs appear similar in their spreading of funnels and gates. The flow field seems to drift toward the east, causing an overall shift in *x*-positions to the right.

[78] In the case of Gauss2, the 10 best designs identified are less wide and considerably more distant from the source zone than in the case of Gauss1. Again, they differ in actual position, but for the *x*-coordinate, at least, this difference is less pronounced. The pattern of a large central barrier with relatively closely spaced surrounding gates seems to hold in the majority of the Gauss2 cases. The flow field of Gauss2 apparently focuses the stream tubes coming from the source zone at some distance down-gradient, where the near-optimal FGS are placed. This could explain the fact that all designs are narrower than the source zone width.

[79] For Gauss3 the widest near-optimal FGS designs were obtained. All of them are positioned closely to the source zone and only minor shifts are present along the *x*-direction. Again, the pattern of a larger central barrier with the gates grouped closely around it seems to hold in the majority of cases.

[80] Considering the overall picture, it appears that introducing a strong heterogeneity, like a large central barrier, is an advantageous feature for the FGS design. It seems intuitive that most of the stream tubes that are deflected by the barrier are forced into the closely spaced gates at its tips. Large central gates, as used in single gate systems, were not identified as good solutions in this study. In fact,

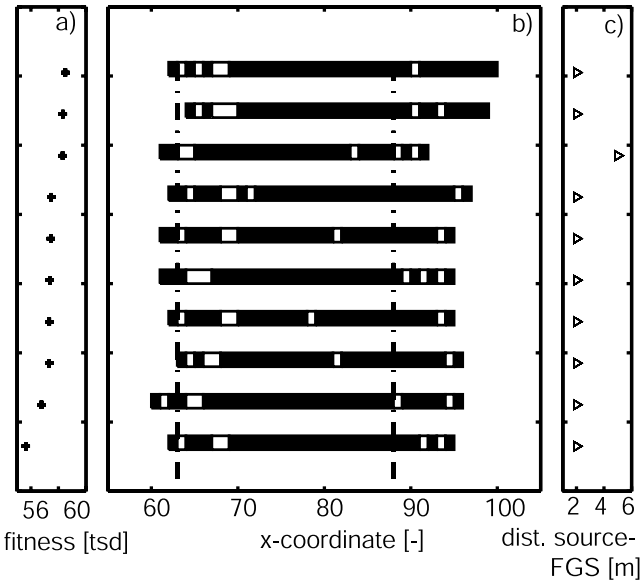

Figure 16. Ten best designs found for Gauss3: (a) fitness and (b) FGS designs and *x*-position (The *x*-position of the source zone is indicated by the two vertical dashed lines.) Please note that only the *x*-direction is scaled. (c) Distance between FGS and source zone.

the way the gates are grouped around the central barrier raised the question of whether these enforced (by the bound constraints on the decision variables) four-gate FGS could be replaced by two-gate FGS in an uncomplicated manner. Starting with the best design solutions, simply replacing the barrier segments between gates (excluding the central barrier) with additional gate cells yielded feasible two-gate FGS for all best solutions of the realizations Gauss1–Gauss3. A further enlargement of the middle barrier at the cost of adjacent gate cells resulted in an even better (lower) fitness in all three cases of two-gate FGSs.

[81] A point of discussion concerning these inferences is that they are only based on investigations covering three different conductivity realizations of the same random function. It could well be that in instances of stronger heterogeneity, where distinct high-conductivity channels dominate the flow regime, a higher number of gates is more beneficial. Along the same line, it is clear that if the head buildup in front of the FGS is also a constraint in the optimization task, larger barriers should be avoided. However, for the optimization task stated, large central funnels seem to provide the most effective control of the flow field.

5. Summary and Conclusions

[82] In this work, a formal procedure was developed for the cost-optimal design of passive remediation technologies, more specifically, funnel-and-gate systems (FGSs). To establish a basis, the main economically relevant technological elements are determined and general cost functions are set up to obtain a quantitative assessment. In principle, these cost functions could be transferred to the conditions of real cases by utilizing site specific values for the individual cost parameters considered, limitations being time-dependent costs or decisions which were not included. The focus was placed on the main technological elements: funnel, gates, and treatment material. In fact, the total cost of such remediation systems can be significantly higher as further site-specific expenditures, such as those for monitoring and maintenance, are ignored when comparing the FGS variants here.

[83] The performance of the FGS is assessed based on its capability to capture and treat a contaminant plume at a hypothetical site implemented in a numerical model. These criteria, as well as cost minimization, were addressed by an objective function, which is revealed to be nonlinear and to exhibit several local optima. These observations have already been made for a simplified one-gate test case, indicating even more complex objective functions for more realistic problems, especially when multigate systems have to be configured. Though working on real-valued decision variables, an up-and-coming variant of evolution strategies, the CMA-ES, achieved stronger convergence properties compared with the binary-coded simple GA applied to the test case. Generally, the results indicated a high reliability in identifying the global optimum for the CMA-ES. Despite the large number of possible FGS variants (>50,000), 70 of 100 CMA-ES runs identified cost-minimized systems. However, alternative implementations of genetic algorithms, which commonly are specifically suited to discrete problems, may represent true alternatives. Therefore research in such a direction is encouraged.

[84] A main question was how to encode the FGS design problem in favor of algorithmic needs while maintaining maximum flexibility for design adjustment. Three different parameterizations of the problem, string, middle segment, and tree encoding, were developed with different levels of decision variable decorrelation. For the more computationally demanding four-gate system, several independent CMA-ES runs were conducted for each encoding in order to identify the most favorable option for different heterogeneous aquifer realizations. Compared with the others, tree encoding improved the fitness of the best individual more quickly in the beginning of the evolutionary search. This might be attributed to the lower correlation between decision variables. Optimization with tree encoding, however, also showed a fine-tuning limitation, as indicated by the early stagnation within the 4000 objective function evaluations considered. Most likely, this effect is caused by the impossibility of providing search-stage-dependent, minimum step-sizes for the CMA-ES. As a conclusion, it can be stated that string encoding and MSE appear to converge more slowly but also more consistently.

[85] For each template aquifer, the FGS configuration with the best optimal function value was only detected in a few independent optimization runs. However, in view of the billions of design alternatives possible (in the case investigated, there are $5^4 \times 30^5$ four-gate FGS designs when only considering permutations of funnel and gate widths), it seems reasonable to evaluate the performance of this optimization procedure regarding its ability to find near-optimal technologies. Furthermore, by setting a tolerance of a few percent on the acceptable objective function value, a significant number of near-optimal solutions were identified with slightly higher costs than the most inexpensive FGS design. The near-optimal systems are distinct from the typically implemented FGS designs. The detailed inspection of these revealed the general features of an optimal system, such as the presence of a large central barrier. In at least the circumstances of this study, these findings substantiate the suitability and worth of algorithmic optimization compared with manual or pure empirical technology adaptation, which would hardly have considered such outside-the-box solutions.

Appendix A

[86] The following formulas given by *Hansen and Ostermeier* [2001] define the CMA-ES algorithm through the transition from generation g to $g + 1$ of the parameter vectors (individuals) $\mathbf{x}_1^{(g)},\ldots,\mathbf{x}_\lambda^{(g)} \in \mathbb{R}^n$, evolution paths $\mathbf{p}_c^{(g)}$, $\mathbf{p}_\sigma^{(g)} \in \mathbb{R}^n$, covariance matrix $\mathbf{C}^{(g)} \in \mathbb{R}^{n\times n}$, and general step-size $\sigma^{(g)} \in \mathbb{R}^+$. The $\langle \mathbf{x} \rangle_w^{(0)}$ and $\boldsymbol{\sigma}^{(0)}$ have to be chosen problem-specific, whereas evolution paths and covariance matrix are initialized as $\mathbf{p}_c^{(0)} = \mathbf{p}_\sigma^{(0)} = 0$ and $\mathbf{C}^0 = \mathbf{I}$ (unity matrix), respectively. All vectors are assumed to be column vectors.

$$\mathbf{x}_k^{(g+1)} = \langle \mathbf{x} \rangle_w^{(g)} + \sigma^{(g)} \underbrace{\mathbf{B}^{(g)}\mathbf{D}^{(g)}\mathbf{z}_k^{(g+1)}}_{\sim N\left(0,\mathbf{C}^{(g)}\right)}, \tag{A1}$$

where

$\mathbf{x}_k^{(g)} \in \mathbb{R}^n$ object parameter vector of the kth individual in generation $g + 1$;

$\langle \mathbf{x} \rangle_w^{(g)}$ $:= \left(1 \Big/ \sum_{i=1}^{\mu} w_i\right) \sum_{i=1}^{\mu} w_i \mathbf{x}_{i:\lambda}^{(g)}, w_i \in \mathbb{R}^+$ weighted mean of the μ best individuals of generation g;

$\mathbf{z}_k^{(g+1)} \in \mathbb{R}^+$ for $k = 1,\ldots,\lambda$ and $g = 0, 1, 2, \ldots$ independent realizations of a $(\mathbf{0}, \mathbf{I})$-normally distributed random vector. Components of $\mathbf{z}_k^{(g+1)}$ are independently (0, 1)-normally distributed.

$\mathbf{C}^{(g)}$ symmetrical positive definite $n \times n$ matrix. $\mathbf{C}^{(g)}$ is the covariance matrix of the normally distributed random vector $\mathbf{B}^{(g)}\mathbf{D}^{(g)}\mathbf{z}_k^{(g+1)}$. $\mathbf{C}^{(g)}$ determines $\mathbf{B}^{(g)}$ and $\mathbf{D}^{(g)}$. $\mathbf{C}^{(g)} = \mathrm{B}^{(g)}\mathbf{D}^{(g)}(\mathbf{B}^{(g)}\mathbf{D}^{(g)})^T = \mathbf{B}^{(g)}(\mathbf{D}^{(g)})^2(\mathbf{B}^{(g)})^T$, which is a singular value decomposition of $\mathrm{C}^{(g)}$.

$\mathbf{D}^{(g)}$ $n \times n$ – diagonal matrix (step size matrix), where the diagonal elements are square roots of eigenvalues of the covariance matrix $\mathbf{C}^{(g)}$, and all off-diagonal elements are zero;

$\mathbf{B}^{(g)}$ orthogonal $n \times n$ matrix (rotation matrix) that determines the coordinate system, where scaling with $\mathbf{D}^{(g)}$ takes place. Columns of $\mathbf{B}^{(g)}$ are (defined as) normalized eigenvectors of the covariance matrix $\mathbf{C}^{(g)}$.

$$\mathbf{p}_c^{(g+1)} = (1 - c_c) \cdot \mathbf{p}_c^{(g)} + c_c^u \cdot \underbrace{c_w \mathbf{B}^{(g)}\mathbf{D}^{(g)}\langle \mathbf{z} \rangle_w^{(g+1)}}_{= \frac{c_w}{\sigma^{(g)}}\left(\langle \mathbf{x} \rangle_w^{(g+1)} - \langle \mathbf{x} \rangle_w^{(g)}\right)} \tag{A2}$$

$$\mathbf{C}^{(g+1)} = (1 - c_{\mathrm{cov}}) \cdot \mathbf{C}^{(g)} + c_{\mathrm{cov}} \cdot \mathbf{p}_c^{(g+1)}\left(\mathbf{p}_c^{(g+1)}\right)^T \tag{A3}$$

where

$\mathbf{p}_c^{(g+1)} \in \mathbb{R}^n$ sum of weighted differences of points $\langle \mathbf{x} \rangle_w$ (from g and $g + 1$);

$c_c \in]0, 1]$ determines the cumulation time for $\mathbf{p}_c$, which is roughly $1/c_c$

$c_c^u := \sqrt{c_c(2 - c_c)}$ normalizes the variance of $\mathbf{p}_c$, because $1^2 = (1 - c_c)^2 + c_c^{u^2}$;

$c_w := \dfrac{\sum_{i=1}^{\mu} w_i}{\sqrt{\sum_{i=1}^{\mu} w_i^2}}$ chosen such that under random selection $c_w\langle \mathbf{z} \rangle_w^{(g+1)}$ and $\mathbf{z}_k^{(g+1)}$ have the same variance (and are identically distributed);

$\langle \mathbf{z} \rangle_w^{(g+1)}$ $:= \dfrac{1}{\sum_{i=1}^{\mu} w_i} \sum_{i=1}^{\mu} w_i \mathbf{z}_{i:\lambda}^{(g+1)}$ with $\mathbf{z}_k^{(g+1)}$ from equations (A1). The notation $i{:}\lambda$ denotes the index of the ith best individual from $\mathbf{x}_1^{(g+1)},\ldots,\mathbf{x}_\lambda^{(g+1)}$. The weights are identical with those for $\langle \mathbf{x} \rangle_w^{(g+1)}$.

$c_{\mathrm{cov}} \in [0, 1[$ change rate of the covariance matrix $\mathbf{C}$. For $c_{\mathrm{cov}} = 0$, no change takes place.

[87] The additional adaptation of the global step size $\sigma^{(g)}$ is taking place on a considerably shorter timescale. For the cumulative path length control in the CMA-ES, a "conjugate" evolution path $\mathbf{p}_\sigma^{(g+1)}$ is calculated, where scaling with $\mathbf{D}^{(g)}$ is omitted.

$$\mathbf{p}_\sigma^{(g+1)} = (1 - c_\sigma) \cdot \mathbf{p}_\sigma^{(g)} + c_\sigma^u \cdot c_w \mathbf{B}^{(g)}\langle \mathbf{z} \rangle_w^{(g+1)} \tag{A4}$$

$$\sigma^{(g+1)} = \sigma^{(g)} \cdot \exp\left(\frac{1}{d_\sigma} \cdot \frac{\| \mathbf{p}_\sigma^{(g+1)} \| - \hat{\chi}_n}{\hat{\chi}_n}\right), \tag{A5}$$

where

$\mathbf{p}_\sigma^{(g+1)} \in \mathbb{R}^n$ evolution path not scaled by $\mathbf{D}^{(g)}$;

$c_\sigma \in]0, 1]$ determines the cumulation time for $\mathbf{p}_\sigma$, which is roughly $1/c_\sigma$;

$c_\sigma^u := \sqrt{c_\sigma(2 - c_\sigma)}$

$d_\sigma \geq 1$ damping parameter, determines the possible change rate of $\sigma^{(g)}$ in the generation sequence;

$\hat{\chi}_n = \mathrm{E}[\| N(\mathbf{0}, \mathbf{I}) \|] = \sqrt{2} \cdot \Gamma\left(\dfrac{n+1}{2}\right) / \Gamma\left(\dfrac{n}{2}\right)$ expectation of the length of a $(\mathbf{0}, \mathbf{I})$-normally distributed random vector.

[88] **Acknowledgments.** We gratefully acknowledge financial support for this work from the German Federal Ministry of Education and Research (Bundesministerium für Bildung und Forschung, BMBF, contract 02WR0195), from the German Research Foundation (Deutsche Forschungsgesellschaft, DFG, contract BA 2850/1-2), and from the European Union FP6 Integrated Project AquaTerra (project GOCE 505428).

References

Atwood, D. F., and S. M. Gorelick (1985), Hydraulic gradient control for groundwater contaminant removal, *J. Hydrol.*, *76*, 85–106.

Bäck, T., D. B. Fogel, and Z. Michalewicz (2000), *Evolutionary Computation: Basic Algorithms and Operators*, Inst. of Phys. Publ., Bristol, UK.

Bayer, P., and M. Finkel (2004), Evolutionary algorithms for the optimization of advective control of contaminated aquifer zones, *Water Resour. Res.*, *40*, W06506, doi:10.1029/2003WR002675.

Bayer, P., and M. Finkel (2006a), Life cycle assessment of active and passive groundwater remediation technologies, *J. Contam. Hydrol.*, *83*(3–4), 171–199.

Bayer, P., and M. Finkel (2006b), Optimization of concentration control by evolution strategies: Formulation, application and assessment of remedial solutions, *Water Resour. Res.*, *43*, W02410, doi:10.1029/2005WR004753.

Bürger, C., and M. Finkel (2005), Performance of multiple gate funnel-and-gate systems in three-dimensional, anisotropic, heterogeneous aquifers, *IAHS Publ.*, *298*, 125–135.

Bürger, C., M. Finkel, and G. Teutsch (2003a), Permeable reactive barriers versus "pump and treat": A cost comparison (Reaktionswandsysteme und "Pump-and-Treat"- Ein Kostenvergleich), *Grundwasser*, *8*(3), 169–180.

Bürger, C. M., M. Finkel, and G. Teutsch (2003b), Technical and economic evaluation of multiple gate funnel-and-gate systems under homogeneous and heterogeneous aquifer conditions, *IAHS Publ.*, *277*, 448–455.

Cai, J., and G. Thierauf (1996), Evolution strategies for solving discrete optimization problems, *Adv. Eng. Software*, *25*, 177–183.

Chan Hilton, A. B., and T. B. Culver (2005), Groundwater remediation design under uncertainty using genetic algorithms, *J. Water Resour. Plann. Manage.*, *131*(1), 25–34.

Cirpka, O. A., C. M. Bürger, W. Nowak, and M. Finkel (2004), Uncertainty and data worth analysis for the hydraulic design of funnel-and-gate systems in heterogeneous aquifers, *Water Resour. Res.*, *40*, W11502, doi:10.1029/2004WR003352.

Craig, J. R., A. J. Rabideau, and R. Suribhatla (2006), Analytical expressions for the hydraulic design of continuous permeable reactive barriers, *Adv. Water Resour.*, *29*(1), 99–111.

Culver, T. B., and C. A. Shoemaker (1993), Optimal control for groundwater remediation by differential dynamic programming with quasi-Newton approximations, *Water Resour. Res.*, *29*(4), 823–831.

Dagan, G. (1989), *Flow and Transport in Porous Formations*, Springer, New York.

Deutsch, C. V., and A. G. Journel (1992), *GSLIB: Geostatistical Software Library and User's Guide*, Oxford Univ. Press, New York.

Dougherty, D. E., and R. A. Marryott (1991), Optimal groundwater management: 1. Simulated annealing, *Water Resour. Res.*, *27*(10), 2493–2508.

Gavaskar, A. (1999), Design and construction techniques for permeable reactive barriers, *J. Hazard. Mater.*, *68*, 41–71.

Goldberg, D. E. (1989), *Genetic Algorithms*, Addison-Wesley, Boston, Mass.

Guan, J., and M. M. Aral (1999), Optimal remediation with well locations and pumping rates selected as continuous decision variables, *J. Hydrol.*, *221*(1–2), 20–42.

Hansen, N. (1998), Verallgemeinerte individuelle Schrittweitenregelung in der Evolutionsstrategie. Eine Untersuchung zur entstochastisierten, koordinaten-systemunabhängigen Adaptation der Mutationsverteilung (Generalized individual step-size control in evolution strategies: A study on the derandomized, coordinate system independent adaptation of the mutation distribution), Ph.D. thesis, Tech. Univ. of Berlin, Berlin.

Hansen, N., and A. Ostermeier (2001), Completely derandomized self-adaptation in evolution strategies, *Evol. Comput.*, *9*(2), 159–195.

Hansen, N., S. D. Müller, and P. Koumoutsakos (2003), Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES), *Evol. Comput.*, *11*(1), 1–18.

Harbaugh, A. W. (1990), A computer program for calculating subregional water budgets using results from the U.S. Geological Survey modular three-dimensional ground-water flow model, *U.S. Geol. Surv. Open File Rep.*, *90-392*, 46 pp.

Harbaugh, A. W., E. R. Banta, M. C. Hill, and M. G. McDonald (2000), MODFLOW-2000, the U.S. Geological Survey modular ground-water model—User guide to modularization concepts and the ground-water flow process, *U.S. Geol. Surv. Open File Rep.*, *00-92*, 121 pp.

Herdy, M. (1991), Application of the Evolutionsstrategie to discrete optimization problems, in *Parallel Problem Solving From Nature*, edited by H. P. Schwefel and R. Männer, *Lect. Notes Comput. Sci.*, *496*, 188–192.

Hsiao, C.-T., and L.-C. Chang (2005), Optimizing remediation of an unconfined aquifer using a hybrid algorithm, *Ground Water*, *43*(6), 904–915.

Karatzas, G. P., and G. F. Pinder (1993), Groundwater management using numerical simulation and the outer approximation method for global optimization, *Water Resour. Res.*, *29*(10), 3371–3378.

Landesumweltamt Nordrhein-Westfahlen (LUA), (1998), *Leistungsbuch Altlastensanierung & Flächenentwicklung 1997/1998* (Service book for clean-up operation and land recycling 1997/1998), Landesumweltamt Nordrhein-Westfahlen, Essen, Germany.

Lefkoff, L. J., and S. M. Gorelick (1986), Design and cost analysis of rapid aquifer restoration systems using flow simulation and quadratic programming, *Ground Water*, *24*(6), 777–790.

Lobo, F. (2000), The parameter-less genetic algorithm: Rational and automated parameter selection for simplified genetic algorithm operation, Ph.D. thesis, Univ. Nova de Lisboa, Lisbon, Portugal.

Maier, U., and P. Grathwohl (2006), Numerical experiments and field results on the size of steady state plumes, *J. Contam. Hydrol.*, *85*, 33–52.

Moreno, L., and C.-F. Tsang (1994), Flow channeling in strongly heterogeneous porous media: A numerical study, *Water Resour. Res.*, *30*(5), 1421–1430.

Mulligan, A. E., and D. P. Ahlfeld (1999), Advective control of groundwater contaminant plumes: Model development and comparison to hydraulic control, *Water Resour. Res.*, *35*(8), 2285–2294.

Nunes, L. M., M. C. Cunha, and L. Ribeiro (2004), Groundwater monitoring network optimization with redundancy reduction, *J. Water Resour. Plann. Manage.*, *130*(1), 33–43.

Papadopoulou, M. P., G. F. Pinder, and G. P. Karatzas (2003), Enhancement of the outer approximation method for the solution of concentration-constraint optimal-design groundwater-remediation problems, *Water Resour. Res.*, *39*(7), 1185, doi:10.1029/2002WR001541.

Pollock, D. W. (1994), User's guide for MODPATH/MODPATH-PLOT, Version 3: A particle tracking post-processing package for MODFLOW, the U.S. Geological Survey finite-difference ground-water flow model, *U.S. Geol. Surv. Open File Rep.*, *94-464*, 249 pp.

Ratzlaff, S. A., M. M. Aral, and F. Al-Khayyal (1992), Optimal design of ground-water capture systems using segmental velocity-direction constraints, *Ground Water*, *30*(4), 607–614.

Rechenberg, I. (1994), *Evolutionsstrategie '94*, Frommann-Holzboog, Stuttgart, Germany.

Reed, P., B. Minsker, and D. E. Goldberg (2000), Designing a competent simple genetic algorithm for search and optimization, *Water Resour. Res.*, *36*(12), 1457–1461.

Rubin, Y. (2003), *Applied Stochastic Hydrogeology*, Oxford Univ. Press, New York.

Sedivy, R. A., J. M. Shafer, and L. C. Bilbrey (1999), Design screening tools for passive funnel and gate systems, *Ground Water Monit. Rem.*, *19*(1), 125–133.

Smith, A. E., and D. W. Coit (1997), Penalty functions, in *Handbook of Evolutionary Computation*, edited by T. Bäck et al., sect. C5.2, pp. 1–6, Oxford Univ. Press, New York.

Starr, R. C., and J. C. Cherry (1994), In situ remediation of contaminated ground water: The funnel-and-gate system, *Ground Water*, *32*(3), 465–476.

Teutsch, G., J. Tolksdorff, and H. Schad (1997), The design of in situ reactive wall systems—A combined hydraulical-geochemical-economical simulation study, *Land Contam. Reclam.*, *5*(3), 125–130.

U.S. Environmental Protection Agency (2002), Field applications of in situ remediation technologies: Permeable reactive barriers, *Contract 68-W-00-084*, Off. of Solid Waste and Emergency Response, Washington, DC.

Wang, W., and D. P. Ahlfeld (1994), Optimal groundwater remediation with well location as a decision variable: Model development, *Water Resour. Res.*, *30*(5), 1605–1618.

Yoon, J.-H., and C. A. Shoemaker (1999), Comparison of optimization methods for ground–water bioremediation, *J. Water Resour. Plann. Manage.*, *125*(1), 54–63.

Zheng, C., and P. P. Wang (1999), An integrated global and local optimization approach for remediation system design, *Water Resour. Res.*, *35*(1), 137–148.

Zheng, C., and P. P. Wang (2002), A field demonstration of the simulation optimization approach for remediation system design, *Ground Water*, *40*(3), 258–264.

P. Bayer, C. M. Bürger, and M. Finkel, Center for Applied Geoscience, University of Tübingen, Sigwartstrasse 10, Tübingen 72076, Germany. (claudius.buerger@uni-tuebingen.de)