

# Inverse modeling of groundwater flow using model reduction

P. T. M. Vermeulen and A. W. Heemink

Faculty of Electrical Engineering, Mathematics and Computer Science, Department of Applied Mathematical Analysis, Delft University of Technology, Delft, Netherlands

J. R. Valstar

Netherlands Institute of Applied Geoscience (TNO), National Geological Survey, Utrecht, Netherlands

Received 1 October 2004; revised 8 February 2005; accepted 15 February 2005; published 2 June 2005.

[1] Numerical groundwater flow models often have a very high number of model cells (greater than a million). Such models are computationally very demanding, which is disadvantageous for inverse modeling. This paper describes a low-dimensional formulation for groundwater flow that reduces the computational burden necessary for inverse modeling. The formulation is a projection of the original groundwater flow equation on a set of orthogonal patterns (i.e., a Galerkin projection). The patterns (empirical orthogonal functions) are computed by a decomposition of the covariance matrix over an ensemble of model solutions. Those solutions represent the behavior of the model as a result of model impulses and the influence of a chosen set of parameter values. For an interchangeable set of parameter values the patterns yield a low-dimensional model, as the number of patterns is often small. An advantage of this model is that the adjoint is easily available and most accurate for inverse modeling. For several synthetic cases the low-dimensional model was able to find the global minimum efficiently, and the result was comparable to that of the original model. For several cases our model even converged where the original model failed. Our results demonstrate that the proposed procedure results in a 60% time reduction to solve the groundwater flow inverse problem. Greater efficiencies can be expected in practice for large-scale models with a large number of grid cells that are used to compute transient simulations.

**Citation:** Vermeulen, P. T. M., A. W. Heemink, and J. R. Valstar (2005), Inverse modeling of groundwater flow using model reduction, *Water Resour. Res.*, 41, W06003, doi:10.1029/2004WR003698.

## 1. Introduction

[2] Groundwater modelers are challenged to simulate the natural system by using numerical models. Nowadays, these models consist of large model networks that describe reality in more and more detail. As a consequence, the computational demands are increased, which is especially undesirable for inverse modeling. Here we adjust a set of variables (e.g., parameter values) such that they decrease the difference (objective function value) between the measured and the computed heads (i.e., inverse modeling). Extensive reviews of inverse models in geohydrology are given by *Carrera and Neuman* [1986], *Cooley* [1985], and *Yeh* [1986]. Several techniques compute sensitivities (i.e., first derivatives of heads with respect to a single estimate variable) [*Cooley*, 1985; *Cooley and Hill*, 1992; *Hill*, 1990; *Mehl and Hill*, 2003] to find an approximate gradient of the objective function. Other, more sophisticated and efficient techniques use the adjoint method [*Courant and Hilbert*, 1953; *Townley and Wilson*, 1985] to find the exact gradient for all estimate variables simultaneously. Beside these optimization methods there are different methods

of assigning parameter values. Most common they are applied according to predefined zones as another, recently developed technique applies parameter values according to influence functions (i.e., representer functions) [*Bennett*, 1992; *Valstar et al.*, 2004]. Other recently developed geostatistical techniques generate hundreds of possible solutions with different structures of the estimate variables to address the notion of uncertainty of the solutions [*Delay et al.*, 2001; *Ginn and Cushman*, 1990; *McLaughlin and Townley*, 1996]. It is beyond the scope of this paper to compare the differences between the mentioned techniques, as this paper describes a different approach that could affect the time efficiency of those techniques positively.

[3] Roughly, the time efficiency can be increased by (1) using a more time-efficient solver [*Mehl and Hill*, 2001], (2) applying a coarse grid and/or a locally refined grid [*Mehl and Hill*, 2003; *Wen et al.*, 2003; *Bennett et al.*, 1996], and/or (3) formulating a low-dimensional model that is capable of simulating the important behavior of the original model [*Cazemier et al.*, 1998; *Hoffman Jørgensen and Sørensen*, 2000; *Krysl et al.*, 2001; *Newman*, 1996; *Park and Cho*, 1996]. For groundwater hydrology *Vermeulen et al.* [2004] obtained such a model by selecting a set of patterns (i.e., empirical orthogonal functions (EOF))

that are most representative of the behavior of the original model. Recently, *Delay et al.* [2001] have used EOF analysis to determine the uncertainty of a stochastic inversion method. In this paper, the EOFs can be seen as a description of the subspace where a possible solution of the original model exists. For hydrological models, it appears that in many directions of the original space the solution is constant. The shape of a well drawdown (its direction in space) will not change dramatically for a change in its amplitude: it will not move around the model domain. This phenomenon makes it possible to project the original partial differential equation upon a given set of patterns, in order to create a low-dimensional model. Because often, the number of patterns is small, such a model reduces the computation time needed (with approximately 2–3 orders of magnitude [*Vermeulen et al.*, 2004]). This makes the reduced model suitable for inverse modeling problems [*Park et al.*, 1999], where the model needs to be evaluated recurrently to find the set of variables with the minimal valued objective function.

[4] In our previous work [*Vermeulen et al.*, 2004], the patterns are representative for the model behavior for one particular set of parameter values. In this paper, the patterns are extended such that they include the sensitivity with respect to different sets of parameter values. The set of parameter values can now be sequentially perturbed, while the set of patterns remains constant and yields a reduced model, over and over again, which is accurate for that combination of parameter values. Another advantage of the reduced model is that its adjoint, which is used to obtain the gradient of the objective function, is easily available (in contrast to that of the original model).

[5] To make this paper self-contained, we first describe the method to create a reduced model via the Galerkin projection in section 2. Section 3 then explains the process of inverse modeling using a reduced model. Finally, section 4 describes the performance and the resulting reduction in computation time for a realistic three-dimensional inverse modeling problem.

## 2. Methodology

### 2.1. Formulation of Groundwater Flow

[6] Three-dimensional groundwater flow, with a uniform density and viscosity, can be described by Darcy's law and the equation of continuity. This yields the following partial differential equation (PDE) [*McDonald and Harbaugh*, 1988]:

$$\frac{\partial}{\partial x} \left[ C_x(\alpha) \frac{\partial \phi}{\partial x} \right] + \frac{\partial}{\partial y} \left[ C_y(\alpha) \frac{\partial \phi}{\partial y} \right] + \frac{\partial}{\partial z} \left[ C_z(\alpha) \frac{\partial \phi}{\partial z} \right] - \beta S \frac{\partial \phi}{\partial t} = -\gamma q \quad (1)$$

where  $\alpha$ ,  $\beta$ ,  $\gamma$  are estimate variables (dimensionless),  $C_x$ ,  $C_y$ ,  $C_z$  are the harmonic hydraulic conductances [ $L^2 T^{-1}$ ] along the  $x$ ,  $y$ ,  $z$  direction, respectively (elaborated along the  $x$  direction in section A1),  $\phi$  is the hydraulic head [L],  $S$  is the storage coefficient (dimensionless),  $t$  is time [T], and  $q$  is a fluid source/sink term [ $LT^{-1}$ ]. One way to solve (1) is to use the finite difference approach to discretize the equation for a mesh of grid cells ( $n_m$ ) in space and time and to solve the entire set of  $n_m$  equations.

[7] In many situations, the model becomes more reliable when a value for  $\alpha$ ,  $\beta$ ,  $\gamma$  can be found that reduces the difference between a set of measurements  $\phi^z$  and its corresponding simulated valued  $\phi$ . A quantity that determines this difference is the sum of weighted squared residuals:

$$\mathcal{J} = \frac{1}{n_z} \sum_{i=1}^{n_z} \sum_{k=1}^{n_t} [\phi_i^z(x, y, z, k) - \phi_i(x, y, z, k, \alpha, \beta, \gamma)]^2 \chi_i \quad (2)$$

where  $\mathcal{J}$  is the objective function [ $L^2$ ],  $\phi_i^z(x, y, z, k)$  is the  $i$ th measurement out of  $n_z$  for time step  $k$  located at position  $(x, y, z)$ , and  $\chi_i$  is the weighting factor for measurement  $i$ . There are several ways to find the optimal values for the estimate variables that minimize  $\mathcal{J}$  [*Carrera and Neuman*, 1986; *Cooley*, 1985; *Tarantola*, 1987; *Press et al.*, 1992]. In this paper, we use a reduced model to reduce the computation time of this minimization problem.

### 2.2. Reduced Model

[8] In this section we describe the creation of a model structure that consists of two parameters; a) spatial patterns and b) a reduced model to obtain corresponding coefficients.

#### 2.2.1. Model Structure

[9] Assume that  $\phi$  can be expressed as a linear combination that can be written by the following equation:

$$\hat{\phi}(x, y, z, k, \alpha, \beta, \gamma) = \sum_{i=1}^{n_p} p_i[x, y, z, \phi_j(\mathcal{L}_j)] r_i(\alpha, \beta, \gamma, k) \quad (3)$$

where  $\hat{\phi}$  is the approximated hydraulic head [L],  $p_i$  is the  $i$ th pattern value out of  $n_p$  (dimensionless), and  $r_i$  is the  $i$ th time-dependent coefficient [L]. It should be noticed that  $r_i$  depends on the current values for the estimate variables  $\alpha$ ,  $\beta$ ,  $\gamma$  and  $p_i$  depends on  $\phi$  which is computed for several sets of estimate variables  $\mathcal{L}_j \in \{\alpha_j, \beta_j, \gamma_j\}$ . It seems disadvantageous that the patterns need to be computed in advance, but for the actual model simulation, this only concerns a limited number of coefficients  $r_i$ . For this reason the following ordinary differential equation is defined that describes  $dr/dt$ :

$$\mathcal{F}[C_x(\alpha), C_y(\alpha), C_z(\alpha), \beta S] r - \frac{dr}{dt} = -\gamma \underline{q}. \quad (4)$$

This equation contains a linear function  $\mathcal{F}$  that depends on the system properties of the original model ( $C_x$ ,  $C_y$ ,  $C_z$ ,  $S$ ) and the estimate variables  $\alpha$ ,  $\beta$ . The function describes the internal relation of the coefficients  $r_i$ . The variable  $\underline{q}$  is a reduced forcing term [ $LT^{-1}$ ] comparable to  $q$  (1). For inverse modeling, the function  $\mathcal{F}$  is recomputed for each perturbation of  $\alpha$  and/or  $\beta$ . Thereafter, the actual simulation of  $dr/dt$  and the evaluation of the objective function (2) yield a reduction in CPU time as this type of model has less dimensions than the original model describing (1).

#### 2.2.2. Pattern Identification (EOFs)

[10] Patterns (empirical orthogonal functions) are the eigenvectors of a covariance matrix that is computed from an ensemble of snapshot vectors [*Park and Cho*, 1996; *Park et al.*, 1999; *Hoffman Jørgensen and Sørensen*, 2000; *Vermeulen et al.*, 2004]. Snapshots are specific result vectors  $\phi_i^j$  that are obtained by the original model given a set of estimate variables  $\mathcal{L}_j$ . They are scaled such that  $\|\phi_i^j\| = 1.0$  because the variation that exists within each snapshot is more important than its amplitude [*Newman*,

1996]. The collection of snapshot vectors ( $n_s^i$ ) are then collected in a matrix  $\Phi_j = [\phi_1^j, \phi_2^j, \dots, \phi_{m^j}^j]$ . This matrix contains the behavior of the original model toward model impulses (e.g., wells, recharge, rivers) for a certain set of estimate variables  $\mathcal{L}_j$ . The original model is computed again for different sets of estimate variables and the total collection of snapshot vectors ( $n_s = n_s^1 + n_s^2 + \dots + n_s^j$ ) becomes the matrix  $\mathbf{D} = [\Phi_1, \Phi_2, \dots, \Phi_j]$  with dimension  $[n_m \times n_s]$ .

[11] Instead of solving the eigenvalue problem [Press et al., 1992] for the very high dimensional covariance matrix  $\mathbf{C}_h = \mathbf{D}\mathbf{D}^T$  with dimension  $[n_m \times n_m]$ , the eigenvalue problem is solved for a reduced covariance matrix, defined as  $\mathbf{C}_r = \mathbf{D}^T\mathbf{D}$  with dimension  $[n_s \times n_s]$  [Krysl et al., 2001]. The eigenvectors  $\mathbf{V}$  of  $\mathbf{C}_h$  can be obtained by applying [Golub and van Loan, 1989, p. 430]

$$\mathbf{V} = \mathbf{D}\mathbf{G}\mathbf{\Lambda}^{-\frac{1}{2}}. \quad (5)$$

This equation results from a decomposition of  $\mathbf{D}^T\mathbf{D}$  and obtains the acquired eigenvectors  $\mathbf{V}$  more efficiently and are equal to those decomposed directly from  $\mathbf{C}_h$ . Eventually, a pattern  $\mathbf{p}_i$  becomes the normalized eigenvector  $\|\mathbf{v}_i\| = 1.0$ , and its corresponding relative importance  $\varphi_i$  is given by

$$\varphi_i(\%) = \frac{\lambda_i}{\sum_{j=1}^{n_s} \lambda_j} \cdot 100 \quad (6)$$

where  $\lambda_i$  is the  $i$ th eigenvalue of  $\mathbf{C}_r$  and/or  $\mathbf{C}_h$ . Experiences suggested that in order to achieve reliable results, one should use at least such a number of patterns that the expected variance  $\varphi^e = \sum_{i=1}^{n_p} \varphi_i \geq 99.9\%$ . The collection of pattern vectors  $\mathbf{p}_i$  are then stored within a *projection* matrix  $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{n_p}]$  with dimension  $[n_m \times n_p]$ . This matrix can be interpreted as a description of a high-dimensional coordinate system that is defined by a limited amount of perpendicular axes/patterns ( $\mathbf{P}^T\mathbf{P} = \mathbf{I}$  i.e., orthogonality). It describes/spans the behavior of the model in the state space. This typical information is used to construct a numerical model (4) that can operate only within the numerical space, described by the set of patterns.

### 2.2.3. Reduced Model via a Galerkin Projection

[12] The Galerkin projection method finds an expression for  $\mathcal{F}$  (4) by substituting the hydraulic head  $\phi$  (1) by  $\hat{\phi}$  (3), yielding

$$\sum_{i=1}^{n_p} \left\{ \frac{\partial}{\partial x} \left[ C_x(\alpha) \frac{\partial p_i}{\partial x} \right] + \frac{\partial}{\partial y} \left[ C_y(\alpha) \frac{\partial p_i}{\partial y} \right] + \frac{\partial}{\partial z} \left[ C_z(\alpha) \frac{\partial p_i}{\partial z} \right] \right\} r_i - \beta S \sum_{i=1}^{n_p} p_i \frac{\partial r_i}{\partial t} = -\gamma q. \quad (7)$$

It can be written in matrix notation, whereby the second-order differential of the pattern derivative of space can be computed in advance (more extensively described by Vermeulen et al. [2004, p. 61]), so

$$\mathbf{U}(\alpha)\mathbf{r} - \beta S \mathbf{P} \frac{d\mathbf{r}}{dt} = -\gamma \mathbf{q} \quad (8)$$

where the matrix  $\mathbf{U}$   $[n_m \times n_p]$  is a function of the vector  $\alpha$ , the matrix  $\mathbf{S}$   $[n_m \times n_m]$  and vector  $\mathbf{q}$   $[n_m \times 1]$  are influenced by the vectors  $\beta$ ,  $\gamma$ , respectively. Equation (8) still operates within  $\mathbb{R}^{n_m}$ , although the dimension of  $\mathbf{r}$  is  $n_p$ , so there are more equations (original dimensions) than unknown variables (i.e., more necessary dimensions than variables to

describe the model behavior). All superfluous equations can be eliminated by multiplying (*projecting*) each term in (8) by the projection matrix  $\mathbf{P}^T$ :

$$\underbrace{\mathbf{P}^T\mathbf{U}(\alpha)}_{\mathbf{N}(\alpha)} \mathbf{r} - \underbrace{\mathbf{P}^T\beta S \mathbf{P}}_{\mathbf{M}(\beta)} \frac{d\mathbf{r}}{dt} = -\underbrace{\mathbf{P}^T\gamma \mathbf{q}}_{\mathbf{q}(\gamma)}. \quad (9)$$

This low-dimensional ODE operates within  $\mathbb{R}^{n_p}$  and is solved for each time step  $k$  with an implicit Euler scheme for the time derivative of  $\mathbf{r}$ :

$$\left[ \mathbf{N}(\alpha) - \frac{1}{\Delta t} \mathbf{M}(\beta) \right] \mathbf{r}_k = -\frac{1}{\Delta t} \mathbf{M}(\beta) \mathbf{r}_{k-1} - \mathbf{q}_k(\gamma). \quad (10)$$

The reduced model consists of two time-independent matrices  $\mathbf{N}$  and  $\mathbf{M}$ , both with a low dimension  $[n_p \times n_p]$ , and a low-dimensional time-dependent vector  $\mathbf{q}_k$  with dimension  $[n_p \times 1]$ . For the actual simulation, only  $\mathbf{q}_k$  needs to be obtained recurrently.

### 2.3. Inverse Modeling

[13] The computed coefficients  $r_i$  (10) are used to reconstruct  $\hat{\phi}$  for the measurement locations. This results in an approximate objective function  $\hat{\mathcal{J}}$ :

$$\hat{\mathcal{J}} = \frac{1}{n_z} \sum_{i=1}^{n_z} \sum_{k=1}^{n_t} \left[ \left( \phi_i^z(x, y, z, k) - \hat{\phi}(x, y, z, k, \alpha, \beta, \gamma) \right) \chi_i \right]^2. \quad (11)$$

[14] A popular method to minimize the objective function  $\hat{\mathcal{J}}$  is to compute the gradient  $\nabla \hat{\mathcal{J}}$  for each estimate variable in which  $\hat{\mathcal{J}}$  declines at a certain location in parameter space (i.e., the current values of  $\alpha$ ,  $\beta$ ,  $\gamma$ ). The gradient can be obtained by perturbing the  $n_u$  estimate variables independently (i.e., by means of a finite difference approximation) and calculating the gradient of the objective function. This requires  $n_u + 1$  normal simulations with the model (i.e., a forward run). However, the gradient can be most efficiently obtained by the adjoint method [Courant and Hilbert, 1953]. This requires one forward run with (10), and one reverse simulation (i.e., one adjoint run) that leads to a reduced adjoint state variable  $\underline{\lambda}$  (dimensionless) (elaborated in Appendix B).  $\nabla \hat{\mathcal{J}}$  is obtained by

$$\begin{aligned} \nabla \hat{\mathcal{J}} &= \frac{\Delta \hat{\mathcal{J}}}{\Delta \alpha} = \sum_{k=1}^{n_t} \underline{\lambda}_k^T \left[ \frac{\partial \mathbf{N}}{\partial \alpha} \mathbf{r}_k \right]; \\ \frac{\Delta \hat{\mathcal{J}}}{\Delta \beta} &= \sum_{k=1}^{n_t} \underline{\lambda}_k^T \left[ -\frac{\partial \mathbf{M}}{\partial \beta} \left( \frac{\mathbf{r}_k - \mathbf{r}_{k-1}}{\Delta t} \right) \right]; \\ \frac{\Delta \hat{\mathcal{J}}}{\Delta \gamma} &= \sum_{k=1}^{n_t} \underline{\lambda}_k^T \left[ \frac{\partial \mathbf{q}_k}{\partial \gamma} \right], \end{aligned} \quad (12)$$

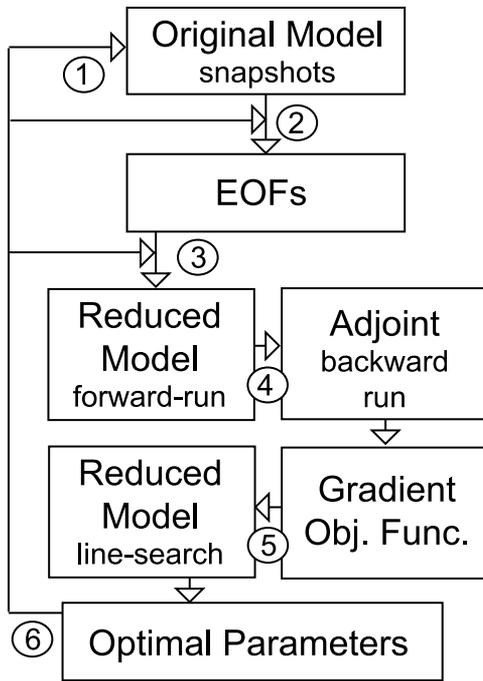
wherein

$$\frac{\partial \mathbf{N}}{\partial \alpha} = \mathbf{P}^T \left[ \frac{\partial}{\partial x} \left( \frac{\partial C_x}{\partial \alpha} \frac{\partial \mathbf{P}}{\partial x} \right) + \frac{\partial}{\partial y} \left( \frac{\partial C_y}{\partial \alpha} \frac{\partial \mathbf{P}}{\partial y} \right) \right. \quad (13)$$

$$\left. + \frac{\partial}{\partial z} \left( \frac{\partial C_z}{\partial \alpha} \frac{\partial \mathbf{P}}{\partial z} \right) \right], \quad (14)$$

$$\frac{\partial \mathbf{M}}{\partial \beta} = \mathbf{P}^T \frac{\partial \mathbf{S}}{\partial \beta} \mathbf{P}, \quad (15)$$

$$\frac{\partial \mathbf{q}_k}{\partial \gamma} = \mathbf{P}^T \frac{\partial \mathbf{q}_k}{\partial \gamma}. \quad (16)$$



**Figure 1.** Methodology for inverse modeling using a reduced model. Numbers refer to steps described in the text.

Notice that the exchange of the variables  $\alpha$ ,  $\beta$ ,  $\gamma$  appears in the dimensions of the original model, prior to the projection with  $\mathbf{P}^T$  (equation (14) is elaborated in section A2). This is disadvantageous for the final efficiency (see section 4.3.3), but currently unavoidable. Once  $\nabla \hat{\mathcal{J}}$  is known, there are various methods to search along that gradient to reach a minimum (i.e., a line search). In this paper we implemented variable metric method (quasi-Newton) [Press *et al.*, 1992]. The method uses information from the gradient to obtain an estimate of the second-order derivative of the objective function (often addressed as the Hessian). This Hessian adjusts the gradient, and from the acquired minimum along that renewed direction, the inverse modeling sequence reiterates. The entire process can be enumerated as follows (Figure 1): (1) The original model is evaluated for specific situations (snapshots) that represent the model behavior and the influence of the estimate variables. (2) From the snapshots a set of orthogonal patterns (EOFs) is computed. (3) A number of patterns is selected to fulfill the expected model variance ( $\varphi^e$ ) and with that set of patterns a reduced model is created. (4) The time-dependent coefficients obtained by one single forward simulation, are then used

for the adjoint run (backward run), to compute the reduced adjoint state variable; this variable provides the gradient of the objective function. (5) The reduced model is simulated several times to search along the obtained gradient toward a minimum of the objective function (i.e., a line search). (6) Finally, it depends on the progress of the inverse modeling whether one needs to proceed to step 3 (gradient loop,  $\eta$ ), 2 (pattern loop,  $\kappa$ ) and/or 1 (snapshot simulation loop,  $\mu$ ).

### 3. Application to a Synthetical Problem

[15] In the following sections the methodology is illustrated by a one-dimensional synthetic model. In section 4 a real-world three-dimensional case is considered.

#### 3.1. Problem Description

[16] A one-dimensional synthetic model was considered that estimated the variable  $\alpha$  (affects the transmissivity) that varies within two predefined zones (Figure 2). The zones were defined by the grid cells  $\mathcal{Z}_1 \in \{1, \dots, 50\}$  and  $\mathcal{Z}_2 \in \{52, \dots, 101\}$ . The well rate varied randomly ( $-50 \leq q \leq 50 \text{ m}^3 \text{ day}^{-1}$ ) for the period  $10 \leq t \leq 1000$  days, and was kept constantly ( $q = -15 \text{ m}^3 \text{ day}^{-1}$ ) for the succeeding period  $1000 < t \leq 2000$  days. The model was simulated with  $\alpha \equiv \beta \equiv \gamma = 1$  and  $\Delta t = 10$  days for each time step, yielding  $n_t = 200$  time steps in total. A set of synthetic measurements were obtained by recording  $\phi$  at all time steps for the grid cells 25 and 76.

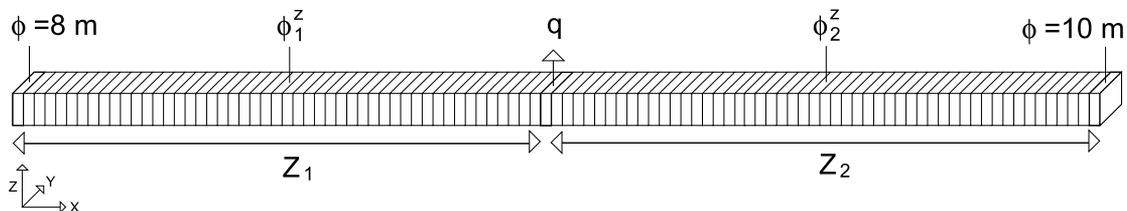
#### 3.2. Snapshot Simulation

[17] In the following section the snapshots are determined for the estimate variable  $\alpha$ . The procedure will be identical for the variables  $\beta$ ,  $\gamma$ .

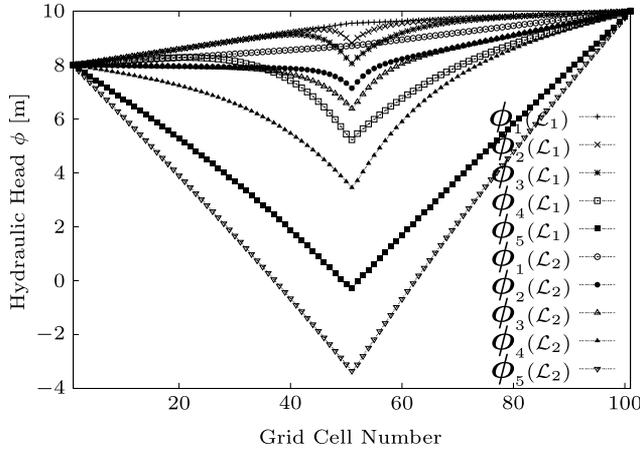
[18] First, the sensitivity of  $\alpha$  can be expressed by simulating the original model with different values for  $\alpha$  [Park and Cho, 1996]. A simple approach is to define a lower and upper boundary for each estimate variable (i.e., snapshot boundary values). These are not the true boundaries of the variable, but they determine a range for which the snapshots are currently representative. For example, these snapshot boundary values can be defined as

$$\alpha_i^S = \exp[\ln(\alpha_i) \pm \delta_i] \quad (17)$$

where  $\delta_i$  is a step size that determines the width of the snapshot boundary for the estimate variable  $\alpha_i$ . The log transformation is applied for reasons of convenience, as the estimate variable  $\alpha_i$  is also log-transformed for the optimization process.



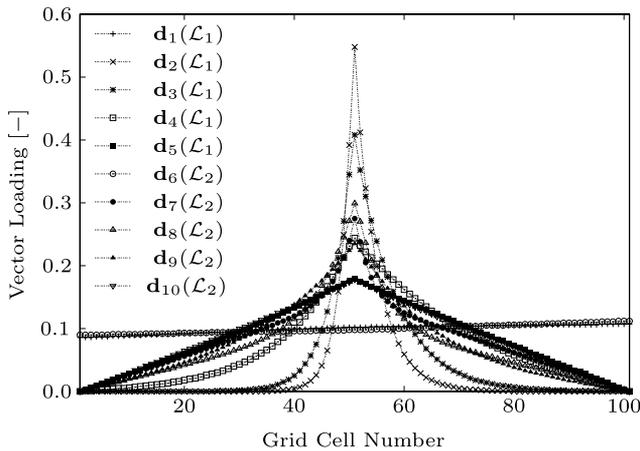
**Figure 2.** Synthetic problem under consideration: 101 grid cells along a single row with an extraction well positioned in the middle and a Dirichlet boundary condition on both edges. Each grid cell is dimensioned by  $\Delta x = \Delta y = 10 \text{ m}$  with  $T = 100 \text{ m}^2 \text{ d}^{-1}$  and  $S = 0.21$ . The model is divided into two zones,  $\mathcal{Z}_1$  and  $\mathcal{Z}_2$ , and two observation wells,  $\phi_1^z$  and  $\phi_2^z$ , with one observation within each zone.



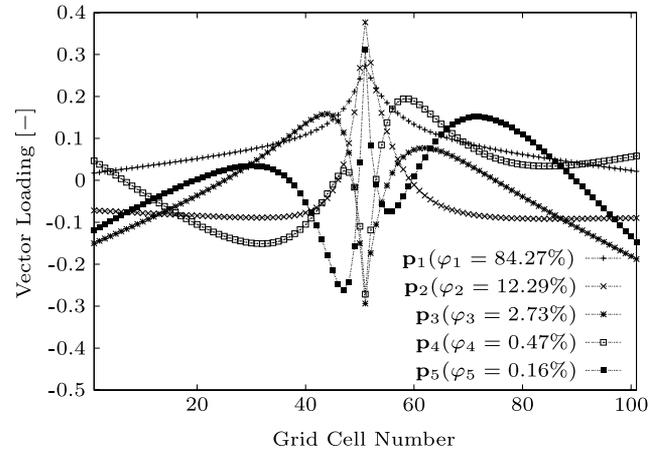
**Figure 3.** Selected snapshot vectors  $\phi_i(\mathcal{L}_j)$  for two combinations of the estimate variable  $\alpha$  computed with the original model.

[19] For the synthetical problem we defined the lower and upper snapshot boundary values as  $\alpha \equiv \delta = 1.0$ . Several snapshot simulations were computed by sequentially perturbing one variable and keeping the others at their lower boundary. This yielded a  $n_u + 1$  set of estimate variables ( $\mathcal{L}$ ) defined by  $\mathcal{L}_1 \in \{\alpha_1^S = 0.36; \alpha_2^S = 0.36\}$ ,  $\mathcal{L}_2 \in \{\alpha_1^S = 0.36; \alpha_2^S = 2.72\}$ ,  $\mathcal{L}_3 \in \{\alpha_1^S = 2.72; \alpha_2^S = 0.36\}$ . Mathematically, this method has a drawback: the pattern identification technique (section 2.2.2) is sensitive to any correlation in the snapshots [Cazemier *et al.*, 1998]. This occurs when a specific variable affects a limited zone and generates almost identical values for  $\phi$  for the remaining zones. The Latin hypercube sampling (LHS) method reduces this correlation [Iman and Shortencarier, 1984]. It divides the range of each estimate variable into  $i$  nonoverlapping intervals on the basis of equal probability. One value is randomly selected from each interval with respect to the probability density in the interval. For the synthetic problem, the LHS method generated  $n_u = 2$  sets of estimate variables valued by  $\mathcal{L}_1 \in \{\alpha_1^S = 0.41, \alpha_2^S = 1.43\}$  and  $\mathcal{L}_2 \in \{\alpha_1^S = 0.94, \alpha_2^S = 0.52\}$ . These were used to compute two snapshot simulations.

[20] A single snapshot simulation  $i$  captured the influence of  $q$  (i.e., the boundary condition and the computation of



**Figure 4.** Adjusted snapshot vectors  $\mathbf{d}_i(\mathcal{L}_j)$  for two combinations of the estimate variable  $\alpha$ .



**Figure 5.** Five major pattern vectors  $\mathbf{p}_i(\varphi^c = 99.927\%)$  with their corresponding relative importance  $\varphi_i$ .

drawdown by a single well) with respect to the variables in  $\mathcal{L}_i$ . A snapshot for the boundary condition was computed as the steady state solution, whereby  $q = 0.0 \text{ m}^3 \text{ day}^{-1}$ . This snapshot  $\phi_1(\mathcal{L}_i)$  represents the gradient through the system caused by the boundary condition on both ends of the model (Figure 3). Thereafter, the extraction rate for the well increased instantaneously up to  $q = 50 \text{ m}^3 \text{ day}^{-1}$  and 4 snapshots were recorded at intermediate time intervals  $t$  ( $t_1 = 10, t_2 = 40, t_3 = 350, t_4 = 5000$  days) until a new steady state situation was reached. The chosen time intervals were not by definition the best as different combinations improved or worsened the reduced model at the end. The most important point is that the record frequency was high initially and decreased as time elapsed. Eventually, these snapshots were captured for each combination of estimate variables defined by  $\mathcal{L}_1$  and  $\mathcal{L}_2$ , yielding  $n_s = n_u \times 5 = 10$  snapshots as indicated in Figure 3. Prior to the pattern identification, the snapshot  $\phi_1(\mathcal{L}_i)$  is subtracted from the corresponding snapshots ( $\phi_j(\mathcal{L}_i); j \in \{2, \dots, 5\}$ ), such that they reflect the influence of the well solely with respect to each combination of variables (see Figure 4).

### 3.3. Resulting Patterns

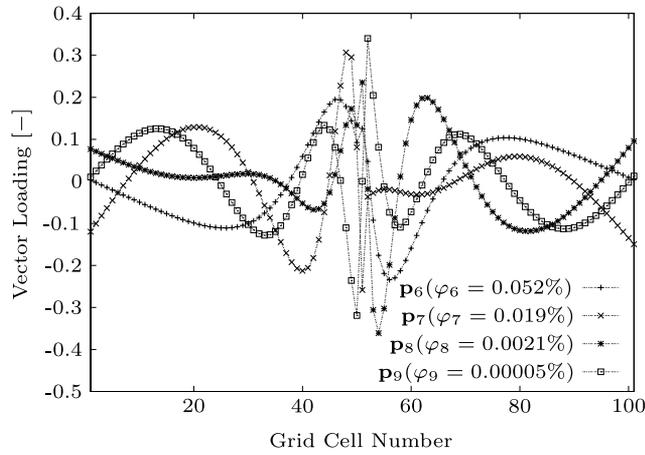
[21] The collection of  $n_s = 10$  snapshots yielded 9 patterns with eigenvalues  $\varphi_i > 0.0\%$ . The maximal explained variance for a pattern is 84.27%, and it declined rapidly as together the five major patterns explained  $>99.9\%$  (Figure 5). Within the spatial structure of  $\mathbf{p}_1$  and  $\mathbf{p}_2$  the solution of  $\phi$  is still recognizable. All other patterns ( $\varphi_i \leq 5\%$ ) describe the numerical space of a solution that does not clearly reflect a hydrological phenomenon (see Figure 6).

### 3.4. Accuracy of $\hat{\mathcal{J}}$

[22] This section describes the sensitivity of the surface of  $\hat{\mathcal{J}}$  with respect to the snapshot simulation and the chosen number of patterns. The section ends with a description of the consequences for inverse modeling.

#### 3.4.1. Surface of $\hat{\mathcal{J}}$

[23] In Figure 7 the surface of the objective function  $\mathcal{J}$  (2) is depicted. The minimum of the objective function ( $\min(\mathcal{J}) = 0.34 \times 10^{-4} \text{ m}^2$ ) is located at  $\ln(\alpha) = 0.0$ . The surface of the approximate objective function  $\hat{\mathcal{J}}$  (11) computed with  $n_p = 9$  patterns is accurate for a significant

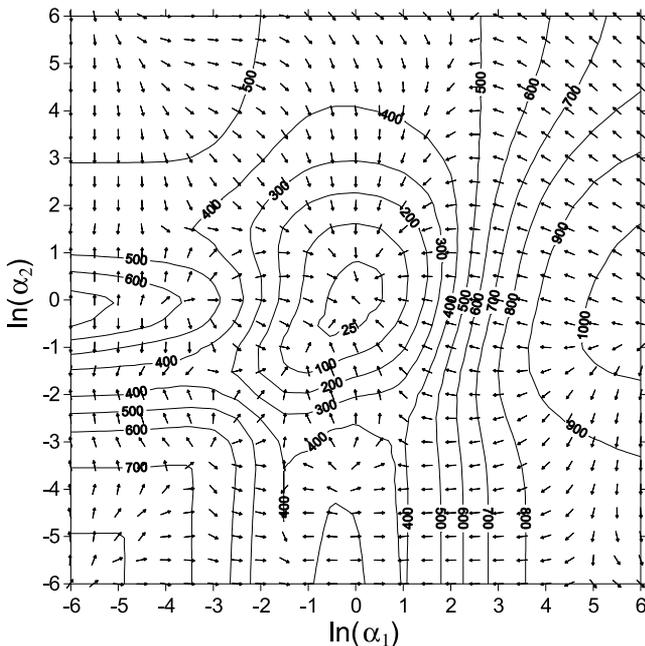


**Figure 6.** Four minor pattern vectors  $\mathbf{p}_i$  ( $\varphi^c = 0.073\%$ ) with their corresponding relative importance  $\varphi_i$ .

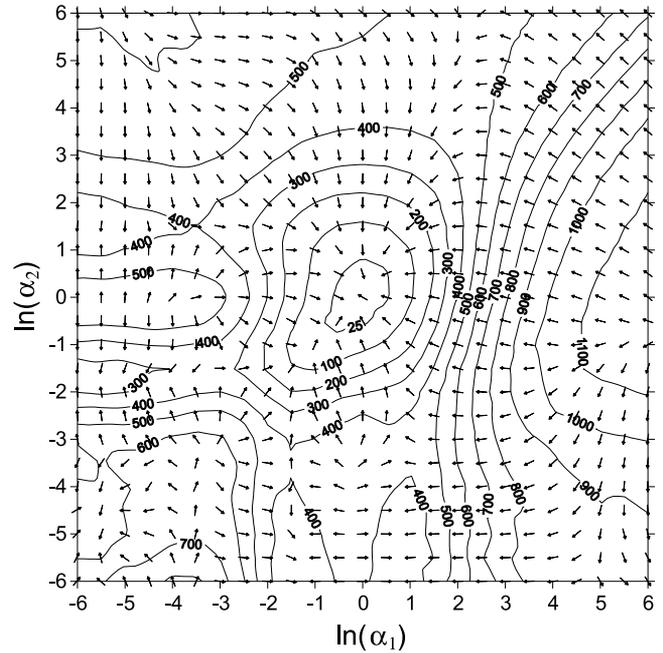
area around the minimum of the function (see Figure 8). Although the minimum value is increased ( $\min(\hat{\mathcal{J}}) = 0.59 \times 10^{-1} \text{ m}^2$ ), its location is almost identical to  $\min(\mathcal{J})$ . It is remarkable that the resemblance between the functions exceeds approximately 3 times the range of the snapshot boundary values for the estimate variables (see section 3.2). When  $\alpha_1, \alpha_2$  were forced toward more extreme values ( $-3.0 \geq \ln(\alpha_i) \geq +3.0$ ), the difference between  $\mathcal{J}$  and  $\hat{\mathcal{J}}$  increased, but the shape of the surface remained mostly similar. This can be seen more clearly in a cross section of the objective function for  $\ln(\alpha_2) = 0.0$  (see Figure 9, lines a and b).

### 3.4.2. Influence of the Snapshot Simulations

[24] Most of the exactness of  $\hat{\mathcal{J}}$  can be traced back to the snapshot simulations (section 3.2) as they form the real bases on which the reduced model is founded. The chosen



**Figure 7.** Surface of the objective function  $\mathcal{J}$  ( $\text{m}^2$ ) for  $\alpha_1, \alpha_2$  computed with the original Modflow model. The arrows represent the gradient  $\nabla \mathcal{J}$ .



**Figure 8.** Surface of the approximate objective function  $\hat{\mathcal{J}}$  ( $\text{m}^2$ ) for  $\alpha_1, \alpha_2$  computed with the reduced model ( $\delta_i = 1$ ,  $\mathcal{L}_1 \in \{\alpha_1^S = 0.41, \alpha_2^S = 1.43\}$  and  $\mathcal{L}_2 \in \{\alpha_1^S = 0.94, \alpha_2^S = 0.52\}$ ,  $n_p = 9$ ,  $\varphi^c = 100\%$ ). The arrows represent the gradient  $\nabla \hat{\mathcal{J}}$ .

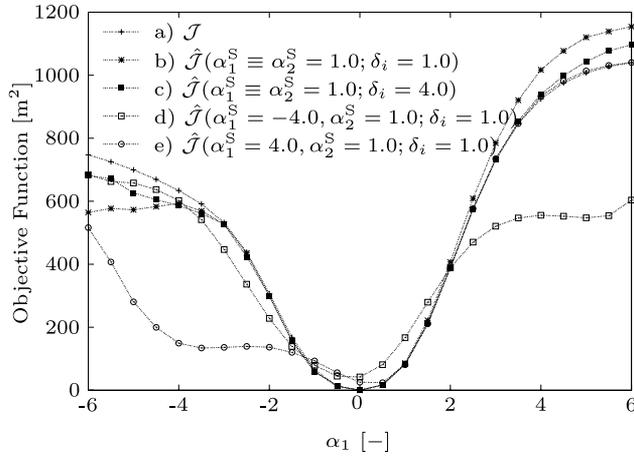
snapshot boundary values (17) are herein most determined as one can see by expanding  $\delta_i = 4.0$ . It will increase the number of patterns and the range of resemblance, but eventually it increases the difference at the minimum, ( $\min(\hat{\mathcal{J}}) = 1.59 \text{ m}^2$ ; see Figure 9, line c). For this synthetic problem this does not seem to be a problem, but numerical experiences for more complex systems with many estimate variables showed that  $\delta_i \geq 2.0$  yielded unreliable objective functions. For such a wide range of parameter values, the assumed linear relation over  $\delta$  is not valid anymore.

[25] It is unlikely that the initial value  $\alpha^0$  is equal to the optimal value. Whenever the initial snapshot simulation ( $\delta_i = 1.0$ ) is computed around  $\ln(\alpha_1) = -4.0$  and  $\ln(\alpha_2) = 0.0$ , the shape of  $\hat{\mathcal{J}}$  is only accurate for the interval  $-6.0 \leq \ln(\alpha_1) \leq 1.0$  (Figure 9, line d). The location of  $\min(\hat{\mathcal{J}})$  has been even shifted in case  $\ln(\alpha_1) = +4.0$ ,  $\ln(\alpha_2) = 0.0$  and  $\delta_i = 1.0$  (Figure 9, line e). These outcomes limit a reduced model to a certain bandwidth to estimate  $\alpha$ . Therefore the reduced model need to be updated after a minimum for the current reduced model is achieved (snapshot simulation loop,  $\mu$ ).

### 3.4.3. Influence of the Number of Patterns

[26] The efficiency of a reduced model increases with a decrease in the number of patterns ( $n_p$ ) [Vermeulen et al., 2004]. Therefore it is a challenge to reduce  $n_p$  such that the resulting approximate objective function is still accurate enough.

[27] As most of the variance occurs near the well, the variance is less near a measurement location. This offers the possibility of neglecting some tiny patterns without severely affecting the shape of the objective function. In Figure 10 the surface of  $\hat{\mathcal{J}}$  is depicted ( $n_p = 5$ ;  $\varphi^c = 99.927\%$ ); thus each pattern in Figure 6 is ignored. The surface shape still

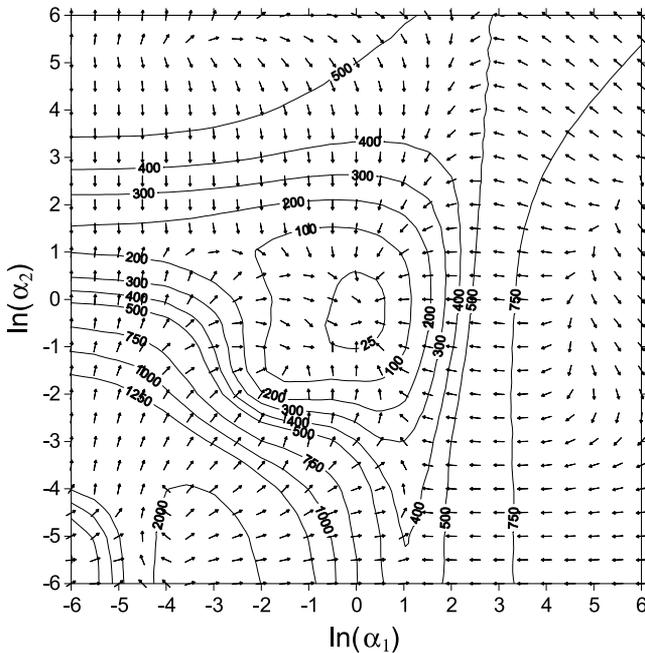


**Figure 9.** Cross section of the objective functions  $\mathcal{J}$  and  $\hat{\mathcal{J}}$  versus  $\alpha_1$  for different values for  $\alpha_1^S$  and  $\alpha_2^S$  that were used to compute snapshots and collect these patterns such that  $\varphi^c = 100\%$ .

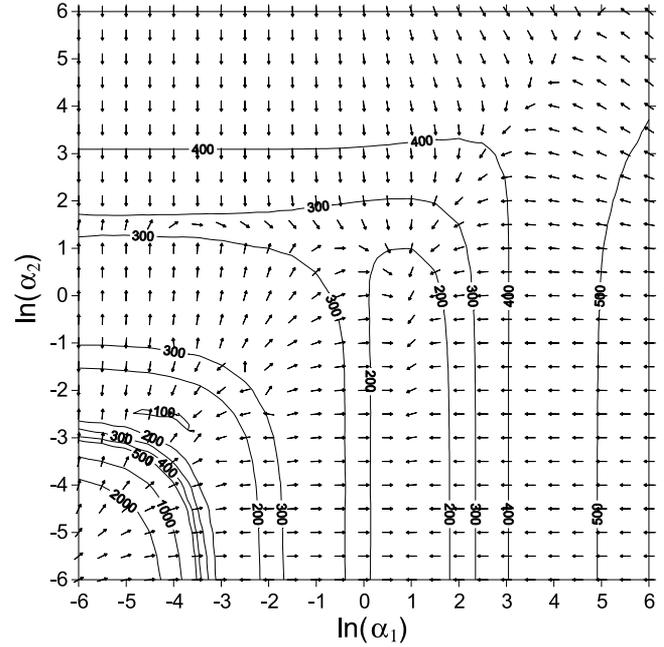
shows a strong resemblance with  $\mathcal{J}$ , but  $\min(\hat{\mathcal{J}}) = 6.5 \text{ m}^2$  and is located at  $\ln(\alpha_1) = 0.095$ ,  $\ln(\alpha_2) = -0.083$ . Nevertheless, with half the number of patterns the optimal value is closely to its final minimum. From this point, more patterns can be added to improve the estimation (pattern loop,  $\kappa$ ). There is a limitation, however, as with the amount of patterns ( $n_p = 3$ ;  $\varphi^c = 99\%$ ) one decreases the accuracy and reliability of the  $\hat{\mathcal{J}}$  (see Figure 11).

#### 3.4.4. Optimization Experiments

[28] This section extends the number of estimate variables for the one-dimensional synthetic problem (section 3.1) to



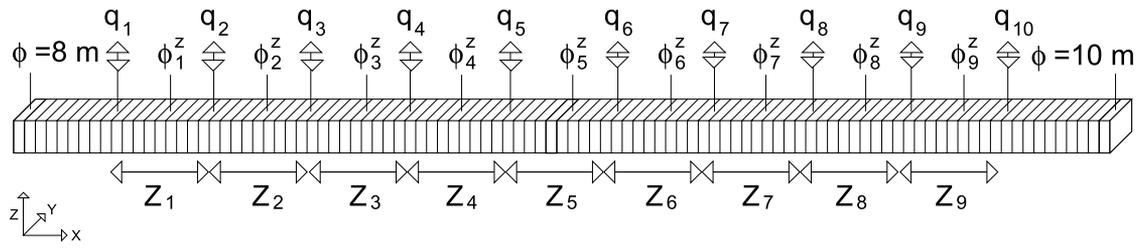
**Figure 10.** Surface of the approximate objective function  $\hat{\mathcal{J}}$  ( $\text{m}^2$ ) for  $\alpha_1, \alpha_2$  computed with the reduced model ( $\delta_i = 1$ ,  $\mathcal{L}_1 \in \{\alpha_1^S = 0.41, \alpha_2^S = 1.43\}$  and  $\mathcal{L}_2 \in \{\alpha_1^S = 0.94, \alpha_2^S = 0.52\}$ ,  $n_p = 5$ ,  $\varphi^c = 99.9\%$ ). The arrows represent the gradient  $\nabla \hat{\mathcal{J}}$ .



**Figure 11.** Surface of the approximate objective function  $\hat{\mathcal{J}}$  ( $\text{m}^2$ ) for  $\alpha_1, \alpha_2$  computed with the reduced model ( $\delta_i = 1$ ,  $\mathcal{L}_1 \in \{\alpha_1^S = 0.41, \alpha_2^S = 1.43\}$  and  $\mathcal{L}_2 \in \{\alpha_1^S = 0.94, \alpha_2^S = 0.52\}$ ,  $n_p = 3$ ,  $\varphi^c = 99\%$ ). The arrows represent the gradient  $\nabla \hat{\mathcal{J}}$ .

9 (see Figure 12). The rate for each well varied randomly and for each observation well a set of measurements was obtained by recording  $\phi$  for all time steps. For 12 test cases (simulations 1–12) the initial estimate variables ( $\alpha_i^0$ ) were varied randomly and corresponding snapshots were computed as described in section 3.2. It yielded  $n_s = 9 \times 4 \times 9 = 324$  snapshots. The optimization procedure as outlined in section 2.3 was used to estimate the optimal value for those estimate variables.

[29] In Table 1 the results of the optimization are given. Both the original and the reduced model succeeded in getting the correct optimal values for the estimate variables (simulations 1–3). Their initial values ( $\alpha_i^0$ ) are within close reach (0.09–10) of their optimal values ( $\|\alpha\| = 1$ ). They needed approximately  $\eta = 9$ –19 gradient iterations. More gradient iterations ( $\eta = 22$ –32) were necessary for the reduced model as the reach of  $\alpha_i^0$  increases (simulations 4 and 5). For specific values of  $\alpha_i^0$  the original model was more sensitive to local minimum, as the reduced model succeeded in finding the global minimum (simulations 6–8). In this case, the reduced model has the benefit of the lack of detail in the objective function. Another example is given in simulation 9, where both models fail to find the global minimum. Whenever the reduced model starts the inverse modeling for this case, by rejecting initially the minor patterns (i.e., their lumped contribution to the explained model variance is less than 1%), it proceeds eventually toward the global minimum, see simulation 10. Unfortunately, this is not a rule of thumb as this strategy can work the other way around, compare simulations 11 and 12. In the end, it can be said that the reduced model is capable of finding an optimal solution for realistic values of  $\alpha$  (0.01–100.0). Beyond that, the method may fail, just like the



**Figure 12.** Synthetic problem under consideration: 101 grid cells along a single row with a Dirichlet boundary condition on both edges. Each grid cell is dimensioned by  $\Delta x = \Delta y = 10$  m with  $T = 100$  m<sup>2</sup> d<sup>-1</sup> and  $S = 0.21$ . The model is divided into nine zones ( $Z_1 - Z_9$ ), each containing an observation well ( $\phi_i^z$ ) and an injection and/or extraction well ( $q_i$ ).

original model, but it may also succeed although the procedure is more ambiguous.

### 4. Application to a Real-World Case

#### 4.1. Introduction

[30] This section examines to what extent the reduced model can reduce the computation time needed for inverse modeling. For this purpose we consider a three-dimensional problem was considered with a realistic parameterization based on a situation in the southern province (Noord-Brabant) in the Netherlands.

#### 4.2. Description of the Original Model

[31] The model area is approximately 30 km<sup>2</sup> translated into  $n_m = 34,224$  grid cells (92 rows and 93 columns) divided over 4 model layers. The model layers 1, 2, and 3 have a closed boundary condition (Neumann boundary). Model layer 4 has an open boundary condition (Dirichlet boundary). The underground is characterized by a detailed distribution of transmissivities that vary between 0–50 (Figure 13), 0–33, 330–540 and 625–1370 m<sup>2</sup> day<sup>-1</sup> for the model layers 1–4, respectively. The resistance for the aquitards in between varies between 15–175, 280–520 and 310–1000 days. The top of the hydrological system is characterized by an area that has an intense surface water system (modeled by a linear relationship between  $\phi$  and

the water level) and a surrounding area with a free-floating water table. The distribution of precipitation was computed by detailed land survey images. There are 10 independent extraction wells, distributed throughout the district (Figure 13). The model was simulated over a period of  $n_t = 200$  time steps ( $\Delta t = 10$  days), and a transient solution is given for model layer 1 for time step 100 in Figure 14. The estimate variable  $\alpha$  was optimized for three zones in model layer 4, for two zones in model layer 3, and for one zone in model layer 2. This yielded 6 estimate variables ( $n_u = 6$ ). Furthermore,  $n_z = 35$  observation wells were selected that were measured for each time step (Figure 14).

#### 4.3. Results

##### 4.3.1. Snapshot Simulation

[32] The driving forces within the model were the boundary conditions (Dirichlet boundary and surface water), precipitation and the extraction wells. To isolate the effects of these driving forces, each snapshot simulation contained a steady state solution with only the boundary conditions. Subsequently a transient impulse response with 9 time steps was computed for the precipitation ( $\Delta t_{k+1} = 2\Delta t_k$  with  $\Delta t_1 = 10$  days), and 10 transient impulse responses were computed for each well, independently ( $\Delta t_1 = 10$ ,  $\Delta t_2 = 20$ ,  $\Delta t_3 = 50$ ,  $\Delta t_4 = 500$  days). This yielded  $1 + 9 + (4 \times 10) = 50$  snapshots for each snapshot simulation. Each of them was carried out with another set of estimate variables

**Table 1.** Several Initial Values for the Estimate Variable  $\alpha$  and the Corresponding Objective Function Value ( $\mathcal{J}^0$ ) and Its Value After  $\eta$  Gradient Iterations of  $\alpha$  for the Original Model ( $\mathcal{J}$ ) and the Reduced Model ( $\tilde{\mathcal{J}}$ )<sup>a</sup>

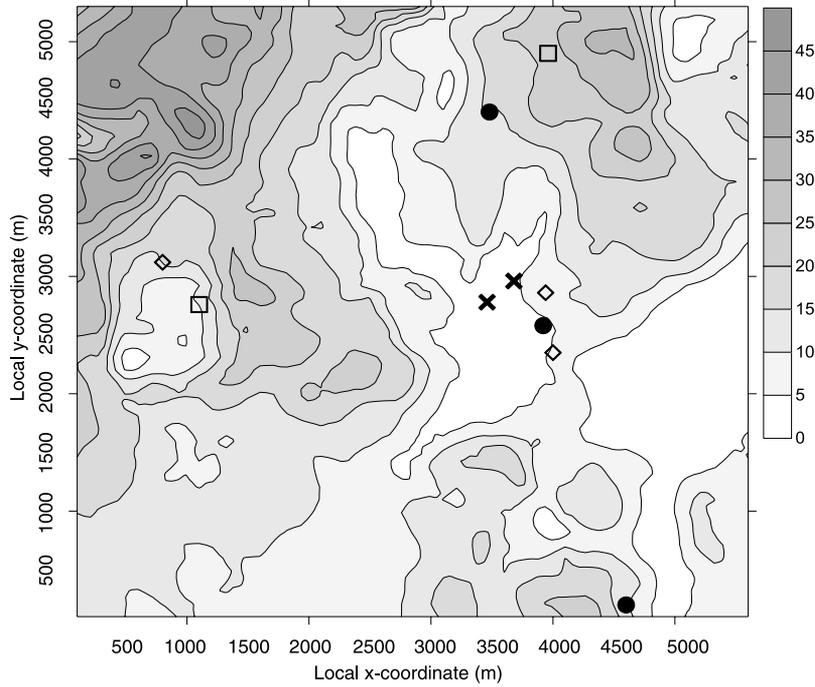
Simulation	Initial Values for the Estimate Variables $\alpha$									$\mathcal{J}^0$ , m <sup>2</sup>	$\mathcal{J}$ , m <sup>2</sup> ( $\eta$ )	$\tilde{\mathcal{J}}$ , m <sup>2</sup> ( $\eta/n_p$ )		
	$\alpha_1^0$	$\alpha_2^0$	$\alpha_3^0$	$\alpha_4^0$	$\alpha_5^0$	$\alpha_6^0$	$\alpha_7^0$	$\alpha_8^0$	$\alpha_9^0$			$\mu = 1$	$\mu = 2$	$\mu = 3$
1	0.098	0.11	0.29	1.70	0.94	2.54	0.06	1.65	0.89	534.22	0.004(10)	0.009(9/40)	-	-
2	1.35	3.39	0.20	2.03	0.08	2.80	0.23	4.31	4.26	101.85	0.006 (12)	0.01(15/35)	-	-
3	0.67	0.76	12.0	0.90	3.74	4.01	10.78	0.84	0.09	48.68	0.008 (12)	0.06(19/35)	-	-
4	15.3	3.97	0.38	0.64	0.10	0.41	1.20	1.60	8.99	44.82	0.008 (26)	10.51(6/34)	0.01 (20/35)	-
5	0.51	0.63	63.0	0.84	9.02	10.1	52.6	0.74	0.02	115.3	0.001 (22)	3.49(20/34)	0.04 (12/32)	-
6	0.17	7.65	0.07	3.24	0.01	5.58	0.09	11.4	11.2	410.46	12.36 (6) <sup>b</sup>	0.30 (20/38)	0.004 (3/37)	-
7	94.1	9.95	0.20	0.48	0.02	0.23	1.37	2.18	38.9	101.89	12.40 (9) <sup>b</sup>	0.69 (20/34)	0.004 (6/36)	-
8	4.45	2.05	102.9	1632	10.96	0.88	373.5	281.7	7.38	66.77	15.29 (22) <sup>b</sup>	10.05 (20/19)	6.67 (10/33)	0.002 (20/34)
9	0.17	0.01	0.11	0.03	5.80	6.72	0.01	14.37	70.2	831.84	15.31 (9) <sup>b</sup>	11.86 (11/41)	7.94 (14/29)	7.70 (4/33) <sup>b</sup>
10	0.17	0.01	0.11	0.03	5.80	6.72	0.01	14.37	70.2	831.84	15.31 (9) <sup>b</sup>	12.03 (14/18) <sup>c</sup>	0.79 (20/17) <sup>d</sup>	0.009 (4/39)
11	3.36	4.28	1066	6.00	103.4	118.7	859.2	5.17	0.06	62.73	13.78 (12) <sup>b</sup>	45.93 (14/10) <sup>c</sup>	9.94 (15/15) <sup>d</sup>	3.12 (20/35) <sup>b</sup>
12	3.36	4.28	1066	6.00	103.4	118.7	859.2	5.17	0.06	62.73	13.78 (12) <sup>b</sup>	5.45 (20/24)	2.82 (7/38)	0.007 (20/37)

<sup>a</sup>The reduced model is based upon a number of patterns ( $n_p$ ) that describe  $\varphi^c = 100\%$  of the model variance, unless stated otherwise.

<sup>b</sup>Local minimum.

<sup>c</sup>Here  $\varphi^c = 99.9\%$ .

<sup>d</sup>Here  $\varphi^c = 99.99\%$ .



**Figure 13.** Distribution of transmissivity ( $\text{m}^2 \text{d}^{-1}$ ) for model layer 1 and location of pumping wells within model layers 1 (circles), 2 (diamonds), 3 (squares), and 4 (crosses).

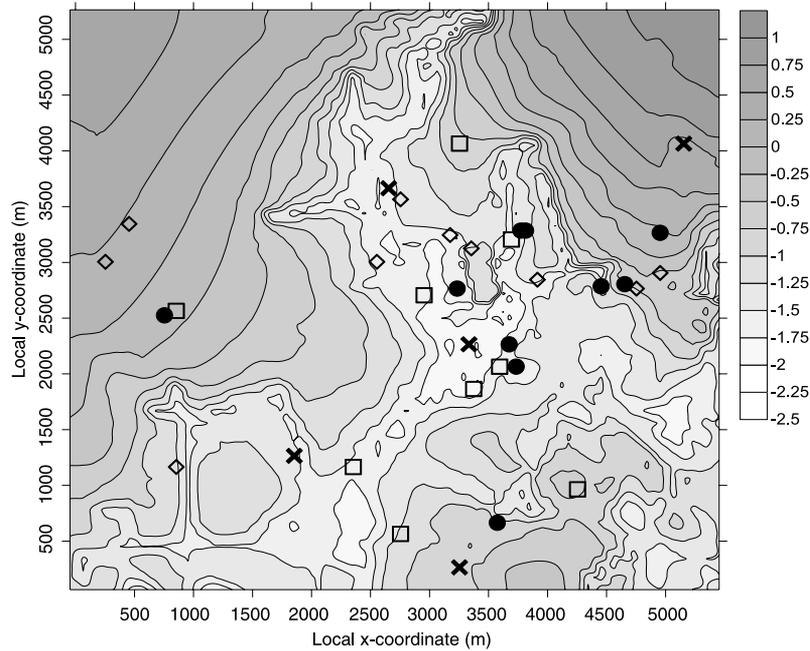
$\mathcal{L}_i \in \{\alpha_1^S, \dots, \alpha_6^S\}$  that was generated by the LHS method (section 3.2). Eventually the total number of snapshots was  $n_s = 6 \times 50 = 300$ , which is 1.5 times the number of total time steps for a single forward model simulation ( $n_t = 200$ ).

**4.3.2. Accuracy of the Estimated Variables**

[33] First, the estimate variables were obtained using the original model. Within  $\eta = 14$  gradient loops, it resulted in a

wide range of estimate variables values (Table 2) with  $\min(\mathcal{J}) = 0.00051 \text{ m}^2$  (see Figure 15).

[34] Second, the estimate variables were obtained by a reduced model (see Figure 16 and Table 2). For the first snapshot loop ( $\mu = 1$ ;  $\delta_i = 1.0$ ) a reduced model was constructed with  $\varphi^c = 99.9\%$ ;  $n_p = 36$  that was able to reduce the objective function and estimate the variables toward the proper direction. After  $\eta = 5$  gradient loops the



**Figure 14.** Computed hydraulic head  $\phi$  (m) for time step 100 and location of observation wells within model layers 1 (circles), 2 (diamonds), 3 (squares), and 4 (crosses).

**Table 2.** Number of Snapshot Simulation Loops ( $\mu$ ) and Gradient Loops ( $\eta$ ) for Inverse Modeling of Six Unknown Estimate Variables  $\alpha$  for the Original Model (OM) and the Reduced Model (RM)

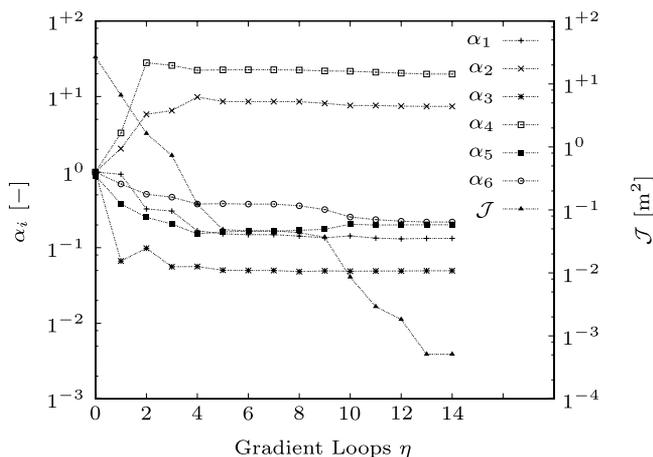
	$\mu/\eta$	$\alpha_1$	$\alpha_2$	$\alpha_3$	$\alpha_4$	$\alpha_5$	$\alpha_6$	$\mathcal{J}, \text{m}^2$
OM	0/0	1.0	1.0	1.0	1.0	1.0	1.0	27.47
OM	0/14	0.13	7.40	0.05	19.80	0.22	0.22	0.00051
RM	0/0	1.0	1.0	1.0	1.0	1.0	1.0	27.34
RM	1/5	0.06	8.46	0.005	3.31	0.08	0.14	8.33
RM	2/10	0.14	6.65	0.05	19.52	0.21	0.21	0.59
RM	3/12	0.14	7.39	0.05	20.10	0.23	0.21	0.024

objective function became stable and another snapshot simulation was performed ( $\mu = 2$ ;  $\delta_i = 1.0$ ) around the current values of the estimate variables. Notice that the objective function was initially enlarged ( $\eta = 6$ ), as the current location in parameter space was not a minimum anymore for the updated reduced model ( $\varphi^c = 99.99\%$ ;  $n_p = 63$ ). Nevertheless, the objective function decreased significantly after  $\eta = 10$  gradient loops, and the estimate variables were close to their optimal values. A final snapshot simulation was carried out ( $\mu = 3$ ;  $\delta_i = 0.5$ ), and the final reduced model ( $\varphi^c = 100.0\%$ ;  $n_p = 103$ ) yielded almost identical values for the estimated variables as found with the original model. As the estimate variables did not change significantly within this third snapshot simulation loop, the inverse modeling was terminated.

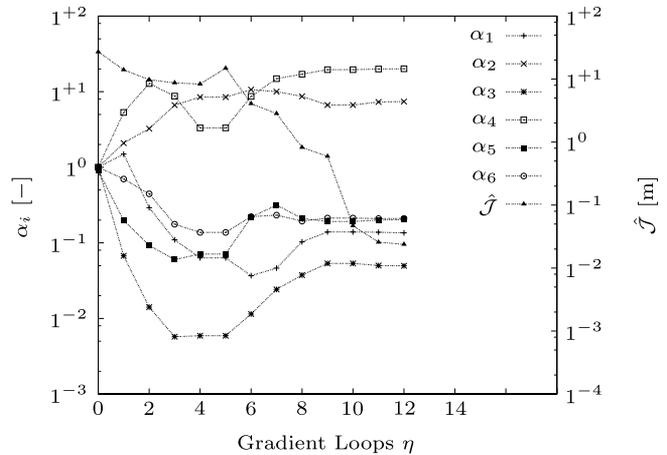
#### 4.3.3. Time Efficiency

[35] All computations for this paper have been carried out on a Pentium 4(R) (2.40 GHz) processor and a single forward or backward run with the original model consumed approximately  $t_o^s = 100.0$  s ( $n_t = 200$ ) (see Figure 17). It took approximately  $t_o^\nabla = 2 \times t_o^s = 200.0$  s to compute  $\nabla\mathcal{J}$ , and the entire parameter optimization with the original model took approximately  $14 \times 200 = 2800$  s.

[36] The time it takes the reduced model to compute  $\nabla\mathcal{J}$  ( $t_r^\nabla$ ) depends on the number of patterns involved. It



**Figure 15.** Graph of the progress of  $\alpha_1, \dots, \alpha_6$  and  $\mathcal{J}$  versus the number of gradient loops  $\eta$  to compute  $\nabla\mathcal{J}$  using the original model.

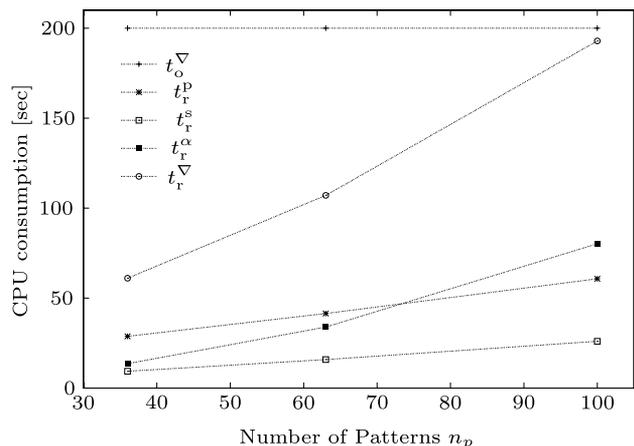


**Figure 16.** Graph of the progress of  $\alpha_1, \dots, \alpha_6$  and  $\hat{\mathcal{J}}$  versus the number of gradient loops  $\eta$  to compute  $\nabla\hat{\mathcal{J}}$  using the reduced adjoint variable  $\underline{\lambda}$ .

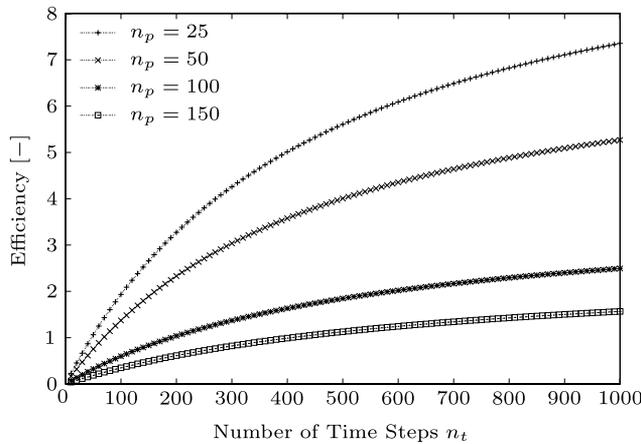
can be calculated as the sum of the time taken to construct a reduced model ( $t_r^p$ ), twice the simulation time of the reduced model (i.e., forward and adjoint run;  $2 \times t_r^s$ ) and the time to compute  $\partial\mathbf{N}/\partial\alpha$  ( $t_r^\alpha$ ) (14). The total time is mostly dominated by  $t_r^\alpha$  that is almost quadratic related to  $n_p$  as it involves two nested “do loops” over  $n_p$  (see Figure 17). As the inverse modeling took  $n_p = 36$  patterns for the first snapshots simulation loop ( $\mu = 1$ ), 63 and 103 for the following snapshot simulation loops ( $\mu = 2, 3$ ), the final computation time can be computed as  $5 \times 60 \text{ s} + 5 \times 100 \text{ s} + 2 \times 195 \text{ s} = 1190 \text{ s}$  (see Table 2 and Figure 18). Hence the total inverse modeling, described in section 4.3.2, resulted in a final time reduction ( $1.0 - t_r^\nabla/t_o^\nabla$ ) of  $1.0 - 1190/2800 \times 100\% \approx 60\%$ .

## 5. Conclusions

[37] This paper describes a new reduced model for inverse modeling of groundwater flow. The proposed model consists of a linear combination of a set of patterns and



**Figure 17.** Graph of the CPU consumption for different processes for inverse modeling versus the number of patterns ( $n_p$ ).



**Figure 18.** Graph of the efficiency, defined as the ratio of computation times required to compute the gradient of the objective function using the original and reduced model for different number of patterns ( $n_p$ ) versus the number of time steps within a simulation.

time-varying coefficients. The patterns (i.e., empirical orthogonal functions) depend both on the response of the hydraulic head  $\phi$  with respect to model impulses (e.g., well, rivers) and system properties (e.g., horizontal/vertical conductances, storage coefficients). These fields of system properties could be defined by zones as done in this paper and/or by stochastic simulation. By selecting the dominant patterns, we have rewritten the original PDE into another one that operates within the numerical space that is described by the selected patterns. Simulating this type of model yields a reduction in CPU time as the number of state variables is small. Since the patterns are based upon different parameter values, the resulting reduced model accurately simulates a wide range for those parameter values. Another advantage is that the adjoint of the reduced model can be obtained fairly easily.

[38] It is not possible to quantify a general rate of efficiency for the proposed procedure, as it depends strongly on the application. For example, the efficiency increases whenever the estimate variable appears to be only the storage coefficient and/or the recharge/well rate. The latter could really benefit the optimization program MODMAN [Greenwald, 1998], which answers groundwater management questions that maximizes or minimizes a user-defined objective function. The efficiency increases even more when the number of time steps in the original model is increased (see Figure 18). This makes this type of inverse modeling highly suitable for long transient simulations. On the other hand, the method is strongly sensitive with respect to the number of patterns (i.e., the complexity of the problem). Such a model may possess many degrees of freedom (e.g., nonlinearity between the hydraulic head and external boundary values and/or model impulses that act independently) that will increase the computation time of the adjoint dramatically. Comparable to other inverse modeling techniques this could be disadvantageous, even though such nonlinearities may not affect the objective function at all.

[39] In future research we intend to focus on the possibility to update the estimate variables in the reduced model.

This will truly increase the final efficiency as it will eliminate the necessity to construct a reduced model each time an estimate variable changes.

## Appendix A: Derivation of the Harmonic Conductances

### A1. Harmonic Conductance

[40] In (1) the harmonic hydraulic conductance  $C_x$  as a function of  $\alpha$  along the  $x$  direction is specified. In this paper a finite difference approximation is used to solve (1), and for a location  $i$  within a mesh of grid cells,  $C_x(\alpha)_i$  is computed as

$$C_x(\alpha)_i = (\Delta x_i + \Delta x_{i+1}) / \left[ \frac{\Delta x_i}{(\alpha T)_i} + \frac{\Delta x_{i+1}}{(\alpha T)_{i+1}} \right] \quad (A1)$$

where  $T$  is the transmissivity [ $L^2T$ ] that is affected by the estimate variable  $\alpha$ . The computation for  $C_y$  is identical for the  $y$  direction.

### A2. First Derivative of the Harmonic Conductance

[41] In (14) the first derivative of the harmonic conductance  $C_x$  with respect to the estimate variable  $\alpha$  is used to compute the second-order differential for the pattern derivative of space. With finite differences this is expressed as

$$\frac{\partial N_{i,k}}{\partial \alpha} = \sum_{m=1}^{n_p} \sum_{j=k}^{n_p} \sum_{i=1}^{n_m} p_{m,i} \Delta x_i^{-1} \cdot \left( \frac{\partial C_{x,i}}{\partial \alpha_i} \frac{p_{j,i} - p_{j,i+1}}{\frac{1}{2} \Delta x_i + \Delta x_{i+1}} - \frac{\partial C_{x,i-1}}{\partial \alpha_{i-1}} \frac{p_{j,i} - p_{j,i-1}}{\frac{1}{2} \Delta x_i + \Delta x_{i-1}} \right) \quad (A2)$$

wherein

$$\frac{\partial C_{x,i}}{\partial \alpha_i} = \frac{\partial C_{x,i}}{\partial T_i} \cdot \frac{\partial T_i}{\partial \alpha_i} + \frac{\partial C_{x,i}}{\partial T_{i+1}} \cdot \frac{\partial T_{i+1}}{\partial \alpha_{i+1}} \quad (A3)$$

using (A1) this yields

$$\frac{\partial C_{x,i}}{\partial \alpha_i} = \frac{\Delta x_i (\Delta x_i + \Delta x_{i+1})}{T_i^2 \left( \frac{\Delta x_i}{T_i} + \frac{\Delta x_{i+1}}{T_{i+1}} \right)^2} \cdot \frac{T_i}{\exp(\alpha_i)} + \frac{\Delta x_{i+1} (\Delta x_i + \Delta x_{i+1})}{T_{i+1}^2 \left( \frac{\Delta x_i}{T_i} + \frac{\Delta x_{i+1}}{T_{i+1}} \right)^2} \cdot \frac{T_{i+1}}{\exp(\alpha_{i+1})} \quad (A4)$$

For the sake of simplicity  $C(\alpha)_{x,i}$  is denoted as  $C_{x,i}$ , and  $\partial T / \partial \alpha$  is equal to  $T / \exp(\alpha)$  as the estimate variable  $\alpha$  is log transformed.

## Appendix B: Derivation of the Reduced Adjoint State Variable

[42] In (12) the reduced adjoint state variable  $\underline{\lambda}$  is introduced in order to compute the gradient of the objective function ( $\nabla \hat{\mathcal{J}}$ ) with respect to the estimate vectors  $\alpha$ ,  $\beta$ ,  $\gamma$ . In order to derive  $\underline{\lambda}$ , we write equation (9) as

$$\mathbf{N}r - \mathbf{M} \frac{dr}{dt} + \underline{q} = 0.0. \quad (B1)$$

Then the objective function  $\hat{\mathcal{J}}$  is defined as

$$\hat{\mathcal{J}} = r^\bullet + \underline{\lambda}^\top \left[ \mathbf{N}\mathbf{r} - \mathbf{M} \frac{d\mathbf{r}}{dt} + \mathbf{q} \right] \quad (\text{B2})$$

where  $r^\bullet$  [ $L^2$ ] represents the sum of the squared residuals between  $n_z$  measurements  $\phi^z$  and  $\hat{\phi}$ :

$$r^\bullet = \sum_{i=1}^{n_z} \left[ (\phi_i^z - \hat{\phi}_i) \chi_i \right]^2 \quad (\text{B3})$$

with

$$\hat{\phi}_i = \sum_{j=1}^{n_p} p_j r_j \quad (\text{B4})$$

This definition of  $r^\bullet$  is necessary to include the state variable  $r$  of the reduced model in the following expansion. To find an expression for  $\Delta \hat{\mathcal{J}}$ , a first-order Taylor series expansion is applied for the variables  $\underline{\lambda}$ ,  $\mathbf{r}$  and  $\alpha$  in (B2), yielding

$$\begin{aligned} \frac{\partial \hat{\mathcal{J}}}{\partial \hat{\mathcal{J}}} \Delta \hat{\mathcal{J}} &= \underline{\lambda}^\top \left[ \underbrace{\mathbf{N}\mathbf{r} - \mathbf{M} \frac{\partial \mathbf{r}}{\partial t} + \mathbf{q}}_{=0} \right] + \frac{\partial \underline{\lambda}^\top}{\partial \underline{\lambda}^\top} \Delta \underline{\lambda}^\top \left[ \underbrace{\mathbf{N}\mathbf{r} - \mathbf{M} \frac{\partial \mathbf{r}}{\partial t} + \mathbf{q}}_{=0} \right] \\ &+ \underline{\lambda}^\top \left[ \underbrace{\mathbf{N}\mathbf{r} - \mathbf{M} \frac{\partial \mathbf{r}}{\partial t} + \mathbf{q}}_{=0} \right] + \frac{\partial r^\bullet}{\partial \mathbf{r}} \Delta \mathbf{r} \\ &+ \underline{\lambda}^\top \left[ \underbrace{\mathbf{N}\mathbf{r} - \mathbf{M} \frac{\partial \mathbf{r}}{\partial t} + \mathbf{q}}_{=0} \right] + \underline{\lambda}^\top \left[ \mathbf{N} \frac{\partial \mathbf{r}}{\partial \mathbf{r}} \Delta \mathbf{r} - \mathbf{M} \frac{\partial}{\partial \mathbf{r}} \frac{\partial \mathbf{r}}{\partial t} \Delta \mathbf{r} + \mathbf{q} \right] \\ &+ \underline{\lambda}^\top \left[ \frac{\partial \mathbf{N}}{\partial \alpha} \Delta \alpha \mathbf{r} - \frac{\partial \mathbf{M}}{\partial \beta} \Delta \beta \frac{\partial \mathbf{r}}{\partial t} + \frac{\partial \mathbf{q}}{\partial \gamma} \Delta \gamma \right] \quad (\text{B5}) \end{aligned}$$

simplifies to

$$\begin{aligned} \Delta \hat{\mathcal{J}} &= \frac{\partial r^\bullet}{\partial \mathbf{r}} \Delta \mathbf{r} + \underline{\lambda}^\top \left[ \mathbf{N} \Delta \mathbf{r} - \frac{1}{\Delta t} \mathbf{M} \Delta \mathbf{r} \right] \\ &+ \underline{\lambda}^\top \left[ \frac{\partial \mathbf{N}}{\partial \alpha} \Delta \alpha \mathbf{r} - \frac{\partial \mathbf{M}}{\partial \beta} \Delta \beta \frac{\partial \mathbf{r}}{\partial t} + \frac{\partial \mathbf{q}}{\partial \gamma} \Delta \gamma \right] \quad (\text{B6}) \end{aligned}$$

The reduced adjoint state variable  $\underline{\lambda}^\top$  is solved by letting

$$\frac{\partial r^\bullet}{\partial \mathbf{r}} \Delta \mathbf{r} + \underline{\lambda}^\top \mathbf{N} \Delta \mathbf{r} - \underline{\lambda}^\top \frac{1.0}{\Delta t} \mathbf{M} \Delta \mathbf{r} = 0.0 \quad (\text{B7})$$

which yields a simple expression as  $\Delta \mathbf{r}$  can be removed, and  $\underline{\lambda}^\top$  is solved backward in time by an implicit Euler scheme:

$$\left( \mathbf{N} - \frac{1}{\Delta t} \mathbf{M} \right)^\top \underline{\lambda}_k = -\frac{1}{\Delta t} \mathbf{M} \underline{\lambda}_{k+1} - \frac{\partial r^\bullet}{\partial \mathbf{r}_k} \quad (\text{B8})$$

where

$$\frac{\partial r^\bullet}{\partial \mathbf{r}_k} = \frac{\partial r^\bullet}{\partial \hat{\phi}_k} \cdot \frac{\partial \hat{\phi}_k}{\partial \mathbf{r}_k} \quad (\text{B9})$$

with

$$\frac{\partial r^\bullet}{\partial \hat{\phi}_k} = 2 \sum_{i=1}^{n_z} \chi_i \left( \phi_{i,k}^z - \hat{\phi}_{i,k} \right) \quad (\text{B10})$$

$$\frac{\partial \hat{\phi}_k}{\partial \mathbf{r}_k} = -\sum_{j=1}^{n_p} p_j \quad (\text{B11})$$

yields a  $n_p$ -dimensional vector. An important difference between (10) and (B8) is the transpose sign T on the left-hand side of the equation.

## Notation

$i, j, k, l, m$	counters.
$\alpha, \beta, \gamma$	estimate variables.
$x, y, z$	cartesian coordinate.
$T$	transmissivity.
$C$	harmonic hydraulic conductance.
$S$	storage coefficient.
$t$	time.
$q$	fluid source/sink term.
$\underline{q}$	reduced source/sink term.
$\phi$	hydraulic head.
$\hat{\phi}$	approximated hydraulic head.
$\phi^z$	hydraulic head measurement.
$\chi$	weighting factor.
$\Phi_j$	collection of snapshots corresponding a set of estimate variables.
$\mathbf{D}$	set of snapshots in matrix notation.
$\mathbf{C}_h$	covariance matrix $\mathbf{D}\mathbf{D}^\top$ .
$\mathbf{C}_r$	reduced covariance matrix $\mathbf{D}^\top \mathbf{D}$ .
$\mathbf{G}$	eigenvectors of $\mathbf{C}_r$ .
$\Lambda$	eigenvalues of $\mathbf{C}_r$ and $\mathbf{C}_h$ .
$\mathbf{P}, \mathbf{V}$	set of patterns in matrix notation.
$p_i$	$i$ th pattern in scalar format.
$\underline{\Lambda}$	reduced adjoint state variable.
$\lambda_i$	$i$ th eigenvalue in scalar format.
$\varphi_i$	relative importance of the $i$ th pattern.
$\varphi^e$	expected variance.
$\mathbf{U}$	second-order differential of the pattern derivative of space.
$\mathbf{N}, \mathbf{M}$	matrices involving the reduced model in state space notation.
$r_i$	$i$ th time-dependent coefficient in scalar format.
$r^\bullet$	sum of the projected sum of squared residuals between $n_z$ measurements $\phi^z$ and $\hat{\phi}$ .
$\mathcal{J}$	objective function original model.
$\hat{\mathcal{J}}$	objective function reduced model.
$\mathcal{F}$	linear function.
$\mathcal{Z}_i$	definition of the $i$ th zone.
$\mathcal{L}_i$	definition of the $i$ th set of $n_u$ estimate variables
$\delta_i$	step size to determine the width of the snapshot boundary for the $i$ th estimate variable.
$n_m$	number of grid cells.
$n_t$	number of simulation time steps.
$n_p$	number of patterns.
$n_s$	number of snapshots simulations.
$n_z$	number of hydraulic head measurements.

$n_u$	number of unknown estimate variables.
$\mu$	number of snapshot simulation loops.
$\kappa$	number of pattern loops.
$\eta$	number of gradient loops.
$\nabla$	gradient.
$t_o^s, t_r^s$	simulation time of the original and reduced model.
$t_o^\nabla, t_r^\nabla$	time to compute $\nabla \mathcal{J}$ and $\nabla \hat{\mathcal{J}}$ .
$t_r^p$	time to compute the patterns.
$t_r^\alpha$	time to compute $\partial \mathbf{N} / \partial \alpha$ .

## References

- Bennett, A. F. (1992), *Inverse Methods in Physical Oceanography*, Cambridge Univ. Press, New York.
- Bennett, A. F., B. S. Chua, and L. M. Leslie (1996), Generalized inversion of a global numerical weather prediction model, *Metereol. Atmos. Phys.*, *60*, 165–178.
- Carrera, J., and S. P. Neuman (1986), Estimation of aquifer parameters under transient and steady state conditions: 1. Maximum likelihood method incorporating prior information, *Water Resour. Res.*, *22*, 199–210.
- Cazemier, W., R. W. C. P. Verstappen, and A. E. P. Veldman (1998), Proper orthogonal decomposition and low-dimensional models for driven cavity flows, *Phys. Fluids*, *10*, 1685–1699.
- Cooley, R. L. (1985), A comparison of several methods of solving nonlinear regression groundwater flow problems, *Water Resour. Res.*, *21*, 1525–1538.
- Cooley, R. L., and M. C. Hill (1992), A comparison of three Newton-like nonlinear least-squares methods for estimating parameters of groundwater flow models, in *Computational Methods in Water Resources 9*, vol. 1, *Numerical Methods in Water Resources*, pp. 379–386, Elsevier, New York.
- Courant, R., and D. Hilbert (1953), *Methods of Mathematical Physics*, Wiley-Interscience, Hoboken, N. J.
- Delay, F., A. Buoro, and G. de Marsily (2001), Empirical orthogonal functions analysis applied to the inverse problem in hydrogeology: Evaluation of uncertainty and simulation of new solutions, *Math. Geol.*, *33*(8), 927–949.
- Ginn, T. R., and J. H. Cushman (1990), Inverse methods for subsurface flow: A critical review of stochastic techniques, *Stochastic Hydrol. Hydraul.*, *4*(1), 1–26.
- Greenwald, R. M. (1998), Documentation and user's guide: MODMAN, an optimization module for MODFLOW, version 4.0, HSI GeoTrans, Freehold, N. J.
- Golub, G., and A. van Loan (1989), *Matrix Computations*, 2nd ed., John Hopkins Univ. Press, Baltimore, Md.
- Hill, M. C. (1990), Relative efficiency of four parameter-estimation methods in steady-state and transient ground-water flow models, in *Computational Methods in Subsurface Hydrology, International Conference on Computational Methods in Water Resources*, edited by G. Gambolati et al., pp. 103–108, Springer, New York.
- Hoffmann Jørgensen, B., and J. N. Sørensen (2000), Proper Orthogonal Decomposition and low-dimensional modelling, *ERCOfTAC Bull.*, *46*, 44–51.
- Iman, R. L., and M. J. Shortencarier (1984), A Fortran 77 program and user's guide for the generation of Latin hypercube and random samples for use with computer models, *Tech. Rep. SAND83-2365*, Sandia Natl. Lab., Albuquerque, N. M.
- Krysl, P., S. Lall, and J. E. Marsden (2001), Dimensional model reduction in non-linear finite element dynamics of solids and structures, *Int. J. Numer. Meth. Eng.*, *51*, 479–504.
- McDonald, M. G., and A. W. Harbaugh (1988), A modular three-dimensional finite-difference groundwater flow model, *U.S. Geol. Surv. Open File Rep.*, *83-875*, Book 6, Chap. A1.
- McLaughlin, D., and L. R. Townley (1996), A reassessment of the groundwater inverse problem, *Water Resour. Res.*, *32*, 1131–1161.
- Mehl, S. W., and M. C. Hill (2001), MODFLOW-2000, the U.S. Geological Survey modular ground-water model, user guide to the Link-AMG (LMG) package for solving matrix equations using an algebraic multigrid solver, *U.S. Geol. Surv. Open File Rep.*, *01-177*.
- Mehl, S. W., and M. C. Hill (2003), Locally refined block-centered finite-difference groundwater models, in *Evaluation of Parameter Sensitivity and the Consequences for Inverse Modelling and Predictions*, IAHS Publ., *277*, 227–232.
- Newman, A. J. (1996), Model reduction via the Karhunen-Loève expansion part I: An exposition, *Tech. Rep. T.R.96-32*, Inst. for Syst. Res., College Park, Md.
- Park, H. M., and C. H. Cho (1996), Low dimensional modeling of flow reactors, *Int. J. Heat Mass Transfer*, *39*, 3311–3323.
- Park, H. M., O. Y. Chung, and J. H. Lee (1999), On the solution of inverse heat transfer problem using the Karhunen-Loève Galerkin method, *Int. J. Heat Mass Transfer*, *42*, 127–142.
- Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery (1992), *Numerical Recipes in Fortran: The Art of Scientific Computing*, Cambridge Univ. Press, New York.
- Tarantola, A. (1987), *Inverse Problem Theory*, 2nd ed., Elsevier, New York.
- Townley, L. R., and J. L. Wilson (1985), Computationally efficient algorithms for parameter estimation and uncertainty propagation in numerical models of groundwater flow, *Water Resour. Res.*, *21*, 1851–1860.
- Valstar, J. R., D. B. McLaughlin, C. B. M. Te Stroet, and F. C. van Geer (2004), A representer-based inverse method for groundwater flow and transport applications, *Water Resour. Res.*, *40*, W05116, doi:10.1029/2003WR002922.
- Vermeulen, P. T. M., A. W. Heemink, and C. B. M. Te Stroet (2004), Reduced models for linear groundwater flow models using empirical orthogonal functions, *Adv. Water Res.*, *27*(1), 57–69.
- Wen, X. H., L. J. Durlafsky, and M. G. Edwards (2003), Use of border regions for improved permeability upscaling, *Math. Geol.*, *35*, 521–547.
- Yeh, W. W. G. (1986), Review of parameter identification procedures in groundwater hydrology: The inverse problem, *Water Resour. Res.*, *22*, 95–108.

---

A. W. Heemink and P. T. M. Vermeulen, Department of Applied Mathematical Analysis, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, P.O. Box 5031, 2600 GA Delft, Netherlands. (p.vermeulen@nitg.tno.nl)

J. R. Valstar, Netherlands Institute of Applied Geoscience (TNO), National Geological Survey, Princetonlaan 6, 3584 CB Utrecht, P.O. Box 80015, 3508 TA Utrecht, Netherlands.