WATER RESOURCES RESEARCH, VOL. 42, W09417, doi:10.1029/2005WR004536, 2006

Model inversion of transient nonlinear groundwater flow models using model reduction

P. T. M. Vermeulen,[1] C. B. M. te Stroet,[1] and A. W. Heemink[2]

Received 28 August 2005; revised 7 April 2006; accepted 9 June 2006; published 30 September 2006.

[1] Despite increasing computational resources many high-dimensional applications are impractical for model inversions. In this paper, two methods are presented that are promising for high-dimensional model inversion. The methods draw on proper orthogonal decomposition (POD) and yield reduced models that describe a truncated behavior of the original model. We utilize POD differently for the two methods, which differ in efficiency and implementation. The first method (RIGM) applies a POD to an existing partial differential equation; the second method (RISM) applies it to an autoregressive formulation of a discrete model. Both methods were applied to several synthetic cases and a real-world case with synthetic measurements and by comparing them to classic inverse methodologies (i.e., the method of finite differences and the adjoint method). The two POD methods appeared to be computationally robust and more efficient for a wide range of prior estimates. Moreover, the implementation of an adjoint for the RISM method is easy.

Citation: Vermeulen, P. T. M., C. B. M. te Stroet, and A. W. Heemink (2006), Model inversion of transient nonlinear groundwater flow models using model reduction, *Water Resour. Res.*, *42*, W09417, doi:10.1029/2005WR004536.

1. Introduction

[2] Computing a model that satisfies (invert) a given observation most optimally is known as inverse modeling. It is widely accepted that a model becomes thereby more reliable, though the model should be thoroughly reviewed to ensure the reasonableness of the results. On the other hand, there are an infinite number of objective functions (calibration criterions) to represent the error of calibration, and the computed calibration parameters are largely influenced by how this objective function is formed. Hence the results of the minimization are not unique, but subject to the chosen estimation variables. So, a single model inversion should be embedded into another inversion cycle that varies the calibration variables, e.g. conceptualization, model parameterizations [*Poeter and Hill*, 1997]. Extensive reviews of inverse models in geohydrology are given by *Carrera and Neuman* [1986], *Cooley* [1985], and *Bennett* [1992]. It is beyond the scope of this paper to compare the differences between the techniques, but it is essential for all of them that the model inversions can be solved within a limited amount of time.

[3] With regard to groundwater modeling, 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 locally refined grid [*Mehl and Hill*, 2003; *Wen et al.*, 2003], and/or (3) formulating a reduced model. This paper describes the last item, which falls in the category of spectral methods. It is worthwhile mentioning that all methods can benefit from parallel computing possibilities. For example, in hydrology this is being exploited by *Wu et al.* [2002] and for reactive transport by *Hammond et al.* [2005]. Efforts to couple models are being designed such that they can be partitioned on different processors [*Winter et al.*, 2004].

[4] Model reduction represents the solution to a problem as a truncated series of known basis functions and independent coefficients (Galerkin scheme), thoroughly elaborated by *Newman* [1996]. Roughly speaking, the goal of these methods is to replace the initial data by data that are optimal in terms of storage capacity. We achieve this by suppressing redundant data that exists within multidimensional data sets. It "breaks up" the initial data sets into different components (i.e., the basis functions) that all together represent the original data set again. This can be obtained by multiplying the components with appropriate coefficients. The basis functions are computed using the proper orthogonal decomposition (POD) (also known as principal component analysis (PCA), empirical orthogonal functions (EOFs), or Karhunen Loève decompositions (KL), although the original concept goes back to *Pearson* [1901]). These functions represent the space of interest as "organized" spatial features (coherent structures) that are optimal for reconstructing and modeling a signal [*Sirovich*, 1987; *Holmes et al.*, 1996]. By projecting the original partial differential equation (PDE) upon a truncated series of basis functions (Galerkin projection), it yields a reduced model to compute the corresponding coefficients that is low dimensional since the set of basis functions is often limited. Recently this type of model was introduced in numerical groundwater flow modeling by *Vermeulen et al.* [2004a, 2004b] and has been applied in

[1] TNO Build Environment and Geoscience, Netherlands Geological Survey, Utrecht, Netherlands.

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

0043-1397/06/2005WR004536$09.00

other fields of science [*Cazemier et al.*, 1998; *Hoffman Jørgensen and Sørensen*, 2000; *Park and Cho*, 1996]. *Park et al.* [1999] mentioned its usefulness for model inversion and implemented a reduced adjoint model for the optimization of a heat source function. *Vermeulen et al.* [2005] implemented an approach for linear groundwater flow in which the model inversion is performed in reduced space and the results are then reconstructed to the original space in order to update the reduced model (dual approach). Quite similar approaches were conducted recently in optimal control design [*Atwell and King*, 2004; *van Doren et al.*, 2005] and atmospheric modeling in which the minimization was done in observation space [*Courtier*, 1997]. The latter relates to the ''balancing'' principle of systems as it truncates spaces that have no influence on observations [*Lee et al.*, 2000; *Newman and Krishnaprasad*, 1998].

[5] To make the reduced model even more time efficient, an ''incremental'' approach can be used in which the original model equation is replaced by a reduced incremental system [*Courtier et al.*, 1994; *Lawless et al.*, 2005; *Vermeulen and Heemink*, 2005]. With this system we are able to find the minimum of an objective function by interchanging the estimation variables within the reduced model only. *Vermeulen et al.* [2005] interchanged these estimation variables within the original model and reformulated the reduced model repeatedly. This particular aspect is minimized within the ''new'' methods and yields therefore higher efficiencies. Moreover, the simulation time of the RISM is negligible because of its autoregressive formulation. Another improvement of the proposed methodologies is that they can handle existing nonlinearities that may appear in the original model. This paper describes two different implementations of these incremental approaches for groundwater models (finite differences).

[6] The paper starts by overviewing the classical methodologies for inverse modeling in section 2. Section 3 is devoted to the reduced model methodologies whereby two different implementations of the POD are evaluated in section 4 and 5 and a note is given for their computational efficiency in section 6. In section 7 several synthetic cases are used to express the performance of the methodologies in term of efficiency and robustness. Finally, section 8 addresses conclusions and recommendations.

2. Classic Methodology

2.1. Forward Model

[7] The algorithms in this paper are elaborated for three-dimensional groundwater flow that can be described by:

$$\sum_{i=1}^{3}\frac{\partial}{\partial x_i}\left[C_i(\alpha_j,x)\frac{\partial\phi(x)}{\partial x_i}\right]-S(x)\frac{\partial\phi(x)}{\partial t}-C_{\mathrm{t}}(x,\phi)\phi(x)$$
$$=q(x)-C_{\mathrm{t}}(x,\phi)z_{\mathrm{t}}(x),\qquad(1)$$

where ϕ is the hydraulic head [L] at location x in the computational domain $x \in \Omega$, C_i is the hydraulic conductance [$\mathrm{L}^2\mathrm{T}^{-1}$] aligned to x_i that depends on an estimation variable α_j (Appendix A), S is the storage coefficient (dimensionless), t is the time [T], q is a specified flux [$\mathrm{L}^3\mathrm{T}^{-1}$] for a portion of the model domain $x \in \partial\Omega_{\mathrm{q}}$, and C_{t} [$\mathrm{L}^2\mathrm{T}^{-1}$] is the total sum of linear and/or nonlinear conductances for a portion of the model domain $x \in \partial\Omega_{\mathrm{t}}$ that depends on $\phi(x)$ and $z_{\mathrm{t}}(x)$, where z_{t} is a specific reference level [L].

[8] The PDE (1) can be solved by means of a finite difference discretization for a mesh of grid cells (O) in space and time with boundary conditions $\phi_{\mathrm{b}}(x)$; $x \in \partial\Omega_{\mathrm{b}}$ as done in MODFLOW [*McDonald and Harbaugh*, 1988]. This discretized model can be written in matrix form as

$$\mathbf{A}[\boldsymbol{\alpha}]\boldsymbol{\phi}(t_i)=\mathbf{b}(t_i),\qquad(2)$$

with

$$\mathbf{A}[\boldsymbol{\alpha}]=\mathbf{C}[\boldsymbol{\alpha}]-\frac{1}{\Delta t}\mathbf{S}-\mathbf{C}_{\mathrm{t}}^{\theta}(t_i),\qquad(3)$$

$$\mathbf{b}(t_i)=-\frac{1}{\Delta t}\mathbf{S}\boldsymbol{\phi}(t_{i-1})+\mathbf{q}(t_i)-\mathbf{C}_{\mathrm{t}}^{\theta}\mathbf{z}_{\mathrm{t}}(t_i),\qquad(4)$$

where $\boldsymbol{\phi} \in \mathbb{R}^O$ is a vector of the nodal hydraulic heads, $\mathbf{C}$ is a heptadiagonal coefficient matrix that contains the hydraulic conductances C at the nodes of a spatial network, $\mathbf{S}$ and $\mathbf{C}_{\mathrm{t}}^{\theta}$ are both diagonal matrices for the elements of the network, where the latter depends in case of nonlinearity subject to $\boldsymbol{\phi}$ on an internal iteration cycle θ (i.e., a Picard iteration), $\mathbf{q}$ and $\mathbf{z}$ are both vectors for fluxes and reference levels, respectively.

2.2. Inverse Model

[9] The objective function value J describes the weighed sum of squared residuals so:

$$J\left[\alpha_j\right]^{\eta}=\sum_{i=1}\left[(\mathbf{y}^{\mathrm{o}}-\mathbf{H}\boldsymbol{\phi})^{\mathrm{T}}\mathbf{W}(\mathbf{y}^{\mathrm{o}}-\mathbf{H}\boldsymbol{\phi})\right](t_i),\qquad(5)$$

where J is the objective function value for the ηth iteration subject to the jth estimation variable α_j, $\mathbf{W}$ is the observational weight matrix, $\mathbf{y}^{\mathrm{o}}$ are the observations, and $\mathbf{H}$ is an operator that maps the model field to observation space. In order to minimize J, two classic methods are described concisely and implemented for purposes of comparison with the reduced model methodologies (see section 3).

[10] 1. The finite difference method (FDM) approximates the first derivative of J with respect to the variable α_j [*Cooley*, 1985] so:

$$\left[\frac{\Delta J}{\Delta\alpha_j}\right]^{\eta}\approx\frac{J\left[\alpha_j^{\eta}+\Delta\alpha_j\right]-J\left[\alpha_j^{\eta}\right]}{\Delta\alpha_j}.\qquad(6)$$

The method is easy to implement, but a disadvantage is that its efficiency decreases with the number of αs (see Figure 1a).

[11] 2. A more sophisticated and time-efficient method computes the exact gradient of J with respect to all αs, simultaneously. This method is known as the adjoint method (ADJ) or 4-D variational method [*Courant and Hilbert*, 1953; *Townley and Wilson*, 1985] so:

$$\left[\frac{\Delta J}{\Delta\alpha_j}\right]^{\eta}=\sum_{i=1}\left\{\boldsymbol{\lambda}^{\mathrm{T}}\left(\frac{\partial\mathbf{A}}{\partial\alpha_{\mathbf{j}}}\right)\boldsymbol{\phi}\right\}(t_i),\qquad(7)$$

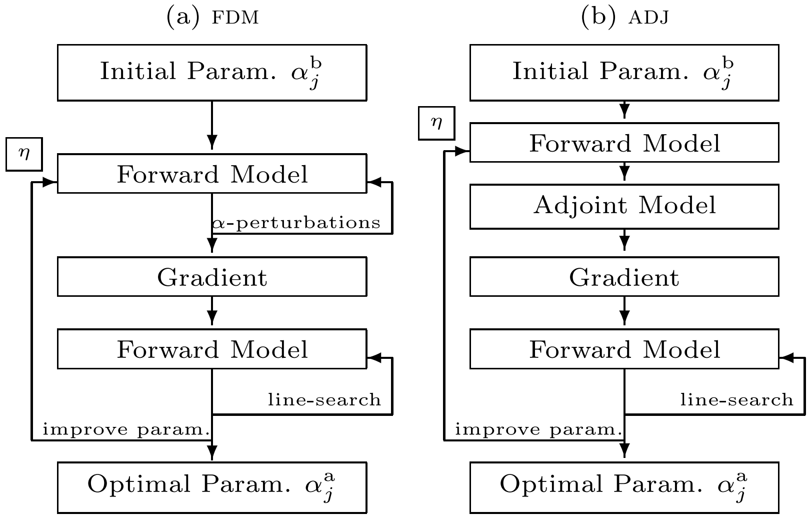


Figure 1. Flowchart of classic inversion methods: (a) the finite difference method (FDM) and (b) the adjoint method (ADJ). The number of gradient computations is expressed by η.

where $\boldsymbol{\lambda}(t_n, t_1)$ is called the adjoint model (concisely derived in appendix B). For each gradient computation it needs to solve the forward model (2) and the adjoint model backwards in time (B3) and is independent from the number of αs (see Figure 1b). However, a disadvantage is that the adjoint model is often hard to implement, especially when the forward model contains nonlinearities.

[12] The classic inversion methods put a lot of effort into a single gradient estimation η, that vanishes completely by the next iteration $\eta + 1$. The proposed methodology in this paper puts this effort into model reduction.

3. Reduced Model Methodology

[13] The reduced model methodology is based upon the Galerkin method. This is a discretization scheme for PDEs that is based upon the assumption that the spatial distribution of the original state vector ($\phi \in \mathbb{R}^{O}$) is very complex in contrast to its behavior in time ($\phi(t_1, t_n)$). For these situations, an approximate state $\hat{\phi}$ [L] can be computed by separating the original state into a spatial and time-variant component:

$$\hat{\boldsymbol{\phi}}(x,t_i) = \boldsymbol{\phi}^{b}(x,t_i) + \mathbf{P}(x)\mathbf{r}(t_i), \tag{8}$$

where $\phi^{b}(x, t_i)$ [L] is the background state, $\mathbf{r}$ [L] is the reduced state equivalent of ϕ, and $\mathbf{P}$ (dimensionless) is a truncated set of spatial basis vectors that span a subspace $\mathcal{S}$ of the original state space $\mathcal{X}$. To acquire this subspace, the original model is simulated several times. The results (i.e., snapshots [*Sirovich*, 1987]) are collected as columns in $\boldsymbol{\Phi}$ and the eigenvectors $\mathbf{p_i}$ of the covariance matrix $\boldsymbol{\Phi}\boldsymbol{\Phi}^{T}$ lead to the projection matrix $\mathbf{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_n\}$, subject to orthonormality. The eigenvalue decomposition can be more efficiently computed from the low-dimensional covariance matrix $\boldsymbol{\Phi}^{T}\boldsymbol{\Phi}$, this is extensively elaborated in *Vermeulen et al.* [2004a, 2004b, 2005]; *Vermeulen and Heemink* [2005].

[14] In this paper we derive different "incremental" models to simulate the coefficients $\mathbf{r}$. We apply a transformation of the original set of equations by using the PODs that yield an ordinary differential equation (ODE) for $\mathbf{r}$ (referred to as "reduced model"; see Figure 2a). These models can then be used, at low computational costs, to optimize a set of estimation variables. This is done by minimizing an approximate objective function $\hat{J}$ defined as:

$$\hat{J}\left[\Delta\alpha_j^{a}\right]^{\zeta} = \sum_{i=1}\left\{\left[\mathbf{P}^{T}\left(\mathbf{y}^{o} - \mathbf{H}\boldsymbol{\phi}^{b}\right) - \mathbf{r}\left(\Delta\alpha_j^{a}\right)\right]^{T}\mathbf{P}^{T}\mathbf{W}\mathbf{P}. \cdot \left[\mathbf{P}^{T}\left(\mathbf{y}^{o} - \mathbf{H}\boldsymbol{\phi}^{b}\right) - \mathbf{r}\left(\Delta\alpha_j^{a}\right)\right]\right\}(t_i), \tag{9}$$

where ζ is the outer iteration index. An advantage is that the minimization of $\hat{J}$ subject to $\Delta\alpha_j^{a}$ can be done with the reduced model only (Figure 2b) and consumes a negligible amount of computation time as $\mathbf{r} \in \mathbb{R}^{n}$ with $n \ll O$. It yields an optimal set of posterior estimates $[\Delta\alpha_j^{a}]^{\zeta}$ for the current ζth reduced model that can still be sub optimal to the original model. Therefore the outer loop $\zeta + 1$ is entered again with $[\alpha_j^{b}]^{\zeta+1} = [\Delta\alpha_j^{a}\alpha_j^{b}]^{\zeta}$ that yields an updated set of snapshots and a different reduced model. This iteration cycle continues until a given termination criterion γ is met for the innovation:

$$\sum_{j=1}\left\|\left[\alpha_j^{b}\right]^{\zeta+1} - \left[\alpha_j^{b}\right]^{\zeta}\right\| \leq \gamma. \tag{10}$$

The statistics that associate the inverse modeling procedure (e.g. residuals, confidence intervals) can be easily computed from the results obtained by a reduced inverse modeling procedure. Each individual residual $\varepsilon_j(t_i)$ can be computed as

$$\varepsilon_j(t_i) = y_j^{o}(t_i) - \sum_{k=1} p_{j,k} r(t_i) \tag{11}$$

where $p_{j,k}$ represents the kth pattern for position j within the model network that corresponds to observation y_j^{o}.

[15] We describe two different formulations based upon (1) a transformation of the governing PDEs (reduced incremental Galerkin model (RIGM)), and (2) a transformation of an autoregressive representation of the governing

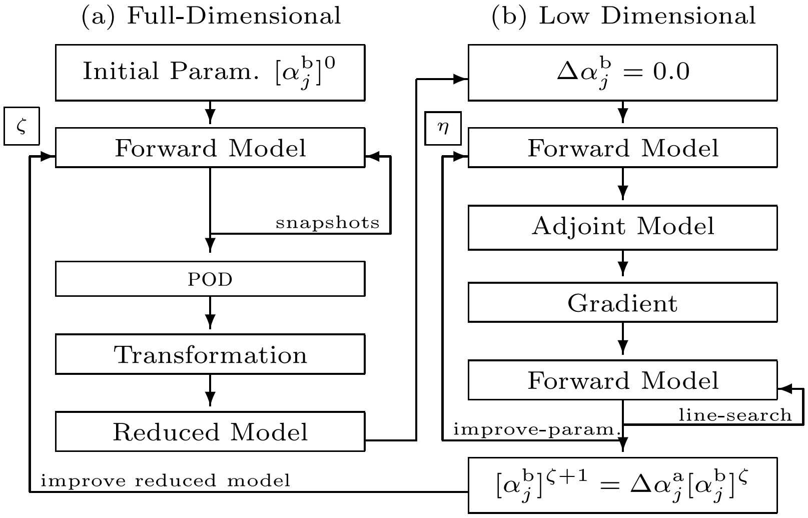


Figure 2. Flowchart of the reduced model methodology for (a) a fully dimensional part to compute PODs that transform the original model into a reduced model and (b) a low-dimensional part used for the model inversion.

system of PDEs (reduced incremental sequential model (RISM)). The two methods differ in their forward and adjoint (backward) model formulations and their snapshots requirements.

4. Reduced Incremental Galerkin Model

4.1. Snapshots

[16] In previous publications [*Vermeulen et al.*, 2004a, 2004b, 2005] effort was put into the computation of snapshots that unravel the model behavior for specific driving forcings subject to α_j^{b}. For simple linear models this can be advantageous, but for nonlinear models this strategy is not sustained and another method is used. Such a method is comparable to the FDM which uses different solutions from the model that each contains the effect of a single parameter perturbation. A large perturbation yields a head derivative to the parameter that is more approximate than the one that is obtained by a small perturbation. The main difference to our snapshots is that they contain several solutions of the model representing each a different combination of parameter perturbations. A large perturbation of parameters yields a reduced model that approximates the objective function more than whenever small perturbations were used. For example, the method defines a snapshot distortion vector $\boldsymbol{\alpha}_i^{\mathrm{s}}$ that affects the parameter T in the original model and yields a snapshot series $\boldsymbol{\Phi}[\boldsymbol{\alpha}_i^{\mathrm{s}}](t_1, t_n)$ for the entire time domain. After that, another vector $\boldsymbol{\alpha}_{i+1}^{\mathrm{s}}$ is defined that yields a different snapshot series (single estimation; see Figure 3a).

[17] From a mathematical point of view, the subsequent vectors $\boldsymbol{\alpha}_i^{\mathrm{s}}$ should be uncorrelated as the eigenvalue decomposition of the snapshots is sensitive to any correlation [*Cazemier et al.*, 1998]. A method that can resolve this is the Latin hypercube sampling (LHS) [*Iman and Shortencarier*, 1984]. This is a constrained Monte Carlo sampling scheme that samples randomly from a probability density function (pdf) and pairs the different samples by minimizing their correlation. The pdf used in this paper is based upon a lognormal distribution of α_j^{b} characterized by a log-transformed deviation factor $\exp(\delta_j)$. Large values for δ_j will increase the subspace $\mathcal{S}$ and yield a reduced model that is less accurate but gives a broad averaged view of $\hat{J}$. Tiny values for δ_j, however, decrease the subspace and compute an accurate $\hat{J}$ for a limited parameter space. In practice, the robustness of the method is improved whenever δ_j is sequentially evolved as $[\delta_j]^{\zeta+1} < [\delta_j]^{\zeta}$.

4.2. Forward Model

[18] The linearization of the high-order model is performed by including an extra term that expresses the dependency of the system dynamics with respect to α_j^{b}

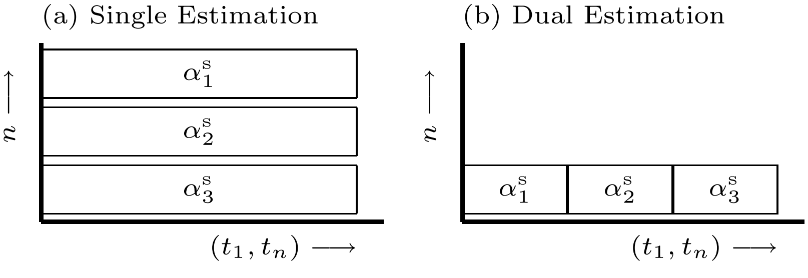


Figure 3. (a) Single- and (b) dual-estimation methodology for the computation of n-snapshot simulations for the time domain (t_1, t_n).

(elaborated in Appendix C). Linearization needs to be performed for a certain distortion vector $\boldsymbol{\alpha}^{\mathrm{b}}$ that yields a specific state, which we call the background state $\phi^{\mathrm{b}}(\boldsymbol{\alpha}^{\mathrm{b}})$ (i.e., fixed point approximation). This is the normal procedure and our adjustment is that we interchange this background state recurrently by the current state ϕ to reflect the current distortion vector $\boldsymbol{\alpha}^{\mathrm{a}}$. This expression is obviously superfluous to the original model, but it yields eventually an incremental reduced model with respect to α_j^{b}. So, including the extra term, substituting the state vector with the Galerkin model (8), and applying a finite difference approximation in space and time, expression (1) becomes:

$$\sum_{i=1}^{3}\left\{\underbrace{\frac{\partial}{\partial x_i}\left[\mathbf{C}_i\frac{\partial\mathbf{P}}{\partial x_i}\right]}_{\mathbf{n}_i}+\sum_{j=1}\underbrace{\frac{\partial}{\partial x_i}\left[\frac{\partial\mathbf{C}_i}{\partial\alpha_j^{\mathrm{b}}}\frac{\partial\mathbf{P}}{\partial x_i}\right]}_{\mathbf{u}_{ij}}\Delta\alpha_j\right\}\mathbf{r}(t_i)$$
$$-\frac{1}{\Delta t}\mathbf{SPr}(t_i)-\mathbf{C}_{\mathrm{t}}^{\theta}\mathbf{Pr}(t_i)=-\frac{1}{\Delta t}\mathbf{SPr}(t_{i-1})$$
$$+\mathbf{q}(t_i)-\mathbf{C}_{\mathrm{t}}^{\theta}\mathbf{z}_{\mathrm{t}}(t_i), \qquad (12)$$

where $\mathbf{C}_i$ is a function of the background variable $[\boldsymbol{\alpha}^{\mathrm{b}}]^{\zeta}$. The expression can be simplified by computing the second-order differentials of the pattern derivative of space and store the results in $\mathbf{N} = \{\mathbf{n}_1, \ldots, \mathbf{n}_n\}$ and $\mathbf{U}_j = \mathbf{u}_{1j}, \ldots, \mathbf{u}_{nj}$. Each term in (12) can now be projected by multiplication with the basis vectors $\mathbf{P}^{\mathrm{T}}$, yielding the reduced equivalents:

$$\mathbf{N}^{\mathrm{r}}=\mathbf{P}^{\mathrm{T}}\mathbf{N}, \qquad (13)$$

$$\mathbf{U}_j^{\mathrm{r}}=\mathbf{P}^{\mathrm{T}}\mathbf{U}_j, \qquad (14)$$

$$\mathbf{S}^{\mathrm{r}}=\mathbf{P}^{\mathrm{T}}\mathbf{SP}, \qquad (15)$$

$$\left(\mathbf{C}_{\mathrm{t}}^{\mathrm{r}}\right)^{\theta}=\mathbf{P}^{\mathrm{T}}\mathbf{C}_{\mathrm{t}}^{\theta}\mathbf{P}, \qquad (16)$$

$$\mathbf{q}^{\mathrm{r}}(t_i)=\mathbf{P}^{\mathrm{T}}\mathbf{q}(t_i), \qquad (17)$$

$$\mathbf{z}_{\mathrm{t}}^{\mathrm{r}}(t_i)=\mathbf{P}^{\mathrm{T}}\mathbf{z}_{\mathrm{t}}(t_i). \qquad (18)$$

It yields the following formulation of the RIGM:

$$\left[\mathbf{N}^{\mathrm{r}}+\sum_{j=1}\mathbf{U}_j^{\mathrm{r}}\Delta\alpha_j-\frac{1}{\Delta t}\mathbf{S}^{\mathrm{r}}-\left(\mathbf{C}_{\mathrm{t}}^{\mathrm{r}}\right)^{\theta}\right]\mathbf{r}(t_i)$$
$$=-\frac{1}{\Delta t}\mathbf{S}^{\mathrm{r}}\mathbf{r}(t_{i-1})+\mathbf{q}^{\mathrm{r}}(t_i)-\left(\mathbf{C}_{\mathrm{t}}^{\mathrm{r}}\right)^{\theta}\mathbf{z}_{\mathrm{t}}^{\mathrm{r}}(t_i), \qquad (19)$$

for which the model is initialized by computing $\mathbf{r}(t_0) = \mathbf{P}^{\mathrm{T}}\phi(t_0)$ and defining a prior incremental estimate $\Delta\alpha_j^{\mathrm{b}}$. The model is nonlinear for the matrix $\mathbf{C}_{\mathrm{t}}^{\mathrm{r}}$, that needs to be evaluated iteratively in the original dimensions and projected upon the basis vectors repeatedly (16). However, this computation needs to be done only for that portion of the model domain $x \in \Omega_{\mathrm{t}}$. For large-scale models it is doubtful whether this iteration is necessary, since due to the projection these slight adjustments within $\mathbf{C}_{\mathrm{t}}^{\mathrm{r}}$ will have a limited influence upon $\hat{J}$. This is a heuristic assumption that increases the efficiency of a reduced model even more. A final remark is that a Dirichlet condition (i.e., a "constant head" cell) should be modeled as a linear boundary condition with a high conductance C such that $\phi(t_i) \approx \mathbf{z}(t_i)$.

4.3. Adjoint Model

[19] An implementation of the adjoint for the reduced model is comparable to that of the original model and eventually yields

$$\frac{\Delta\hat{J}}{\Delta\alpha_j}=\sum_{i=1}\left[\boldsymbol{\lambda}^{\mathrm{r}}(t_i)\right]^{\mathrm{T}}\mathbf{U}_j^{\mathrm{r}}\mathbf{r}(t_i), \qquad (20)$$

where $\boldsymbol{\lambda}^{\mathrm{r}}(t_n, t_1)$ is the reduced adjoint model (dimensionless) (elaborated in Appendix D) and should be computed backwards in time to represent the reduced system dynamics with respect to the partial derivative $\partial\hat{J}/\partial\mathbf{r}(t_i)$. To ensure that $\mathbf{C}_i + (\partial\mathbf{C}_i/\partial\alpha_j^{\mathrm{b}})\Delta\alpha_j$ remains positive during the optimization procedure, $\Delta\alpha_j[\alpha_j^{\mathrm{b}}]^{\eta}$ is log-transformed. As a result, the partial derivative $\partial T_i^{\mathrm{b}}/\partial\alpha_j$ (C1) becomes $[\alpha_j^{\mathrm{b}}]^{\zeta}/\alpha_j T_i^{\mathrm{b}}$ and α_j becomes a part of the reduced PDE (12). As a consequence, the matrix $\mathbf{U}_j^{\mathrm{r}}$ needs to be recomputed for each perturbation of α_j, which will decrease the overall efficiency of the method. We have avoided this by linearizing the partial differential for α_j^{b} so it became $[\alpha_j^{\mathrm{b}}]^{\zeta}T_i^{\mathrm{b}}$. Because of this, the adjoint gradient will become less accurate beyond this "fixed point" and the matrix $\mathbf{U}_j^{\mathrm{r}}$ should be updated during the minimization. A simple procedure is to recompute this matrix only when no further improvement of the objective function is achieved (i.e., ξ iteration). From the synthetic examples, however, the adjoint gradients were still accurate for a reasonable parameter space.

5. Reduced Incremental Sequential Model

5.1. Governing Equation

[20] The method writes the original PDE (1) as an autoregressive model formulation:

$$\phi(t_i)=M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]\phi(t_{i-1})+\sum_{j=1}\frac{\partial M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]}{\partial\alpha_j}\Delta\alpha_j, \qquad (21)$$

where the influence of $\Delta\alpha_j$ upon the dynamics operator M_i is linearized and M_i itself is the system dynamics operator as function of the background estimate $\boldsymbol{\alpha}^{\mathrm{b}}$ defined as:

$$M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]=\tilde{\mathbf{A}}_i^{-1}\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]\tilde{\mathbf{b}}(t_i), \qquad (22)$$

where $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{b}}$ could in fact represent a collection of system dynamics operators influenced by a collection of different linear and nonlinear forcings, respectively (e.g., a coupled surface water and groundwater model). The dynamics operator M_i that we are looking for (22) represents the dynamics and the forcings within the entire set of models. It will be a full matrix with dimension $\sum_{i=1}O_i^2$ that will be impossible to compute for realistic problems. However, in reduced space this matrix can be obtained fairly easy as shown in the following subsection.

5.2. Snapshots

[21] Referred to this RISM method, the usage of snapshots is twofold; they are needed to (1) describe the model behavior, and (2) to represent $\partial M_i[\boldsymbol{\alpha}^{\mathrm{b}}]/\partial\alpha_j$. The latter can be obtained by the classic method of finite differences (6)

that sequentially samples each estimation variable α_j. As mentioned earlier, this is disadvantageous for the quality of the basis vectors **P**, and it is therefore better to perturb all variables simultaneously (section 4.1). As a consequence, $\partial M_i[\boldsymbol{\alpha}^{\mathrm{b}}]/\partial\alpha_j$ cannot be extracted from the snapshots anymore. Hence we introduce $\partial M_i[\boldsymbol{\alpha}^{\mathrm{b}}]/\partial\beta_j$ instead, where the vector $\Delta\boldsymbol{\beta}$ is computed as:

$$\Delta\boldsymbol{\beta} = \hat{\mathbf{Z}}^{\mathrm{T}}\mathbf{Z}\Delta\boldsymbol{\alpha}, \tag{23}$$

where $\hat{\mathbf{Z}} = \{\hat{\mathbf{z}}_1, \ldots, \hat{\mathbf{z}}_n\}$ is a projection matrix and $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_n\}$ is a matrix that maps the estimation variable on the model field [*Vermeulen and Heemink*, 2005]. So, instead of perturbing along $\mathbf{z}_j\Delta\alpha_j$, $\partial M_i/\partial\beta_j$ is computed by perturbing along $\hat{\mathbf{z}}_j\Delta\beta_j$:

$$\frac{\partial M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]}{\partial\beta_j}(t_i) \approx \frac{\boldsymbol{\phi}\left\{\tilde{\mathbf{A}}\left[\boldsymbol{\alpha}_j^{\mathrm{b}} + \Delta\beta_j\hat{\mathbf{z}}_j\right], \tilde{\mathbf{b}}\right\} - \boldsymbol{\phi}\left\{\tilde{\mathbf{A}}\left[\boldsymbol{\alpha}_j^{\mathrm{b}}\right], \tilde{\mathbf{b}}\right\}}{\Delta\beta_j}(t_i), \tag{24}$$

where $\tilde{\mathbf{A}}[\boldsymbol{\alpha}_j^{\mathrm{b}} + \Delta\beta_j\hat{\mathbf{z}}_j]$ is elaborated in Appendix E. From a mathematical point of view, $\Delta\alpha_j$ should be chosen with care as it influences the diagonal dominance of the system matrices **A**. Most iterative solvers should satisfy this condition, which can potentially be satisfied by operator splitting that splits the system matrix into a diagonal dominant part and a remaining part [*Edwards*, 2000]. Finally, the snapshot vectors $\boldsymbol{\Phi}$ will contain the results of $\partial M_i/\partial\beta_j$ for the locations of observation only:

$$\boldsymbol{\Phi} = \left\{\mathbf{H}\frac{\partial M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]}{\partial\beta_j}, \ldots, \mathbf{H}\frac{\partial M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]}{\partial\beta_n}\right\}. \tag{25}$$

This concept is often used for a modal decomposition of input-state-output systems ("balanced realizations" [*Lee et al.*, 2000; *Newman and Krishnaprasad*, 1998]) that disregard those parts of the system that are not observable. As a result, the snapshot vectors describe less variance that yields less patterns, compared to those that were needed to describe the variance for the entire model domain.

5.3. Forward Model

[22] The autoregressive model (21) can be projected by $\mathbf{P}^{\mathrm{T}}$ and denoted as a partitioned matrix

$$\begin{bmatrix}\mathbf{r}\\ \Delta\boldsymbol{\beta}\end{bmatrix}(t_i) = \begin{bmatrix}\mathbf{M}^{\mathrm{r}} & \mathbf{M}^{\beta}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}_i \begin{bmatrix}\mathbf{r}\\ \Delta\boldsymbol{\beta}\end{bmatrix}(t_{i-1}), \tag{26}$$

where

$$\mathbf{M}_i^{\beta} = \mathbf{P}^{\mathrm{T}}\left\{\mathbf{H}\frac{\partial M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]}{\partial\beta_1}, \ldots, \mathbf{H}\frac{\partial M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]}{\partial\beta_n}\right\}, \tag{27}$$

$$\mathbf{M}_i^{\mathrm{r}} = \mathbf{P}^{\mathrm{T}}\mathbf{H}M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]\mathbf{P}. \tag{28}$$

The operator $\mathbf{M}_i^{\mathrm{r}}$ can be rewritten as:

$$\mathbf{M}_i^{\mathrm{r}} = \mathbf{P}^{\mathrm{T}}\left\{\mathbf{H}\frac{\partial M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]}{\partial\boldsymbol{\phi}}\mathbf{p}_1, \ldots, \mathbf{H}\frac{\partial M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]}{\partial\boldsymbol{\phi}}\mathbf{p}_n\right\}(t_i), \tag{29}$$

for which each column j can be approximated by slightly perturbing the function M_i over ϵ along the direction of $\mathbf{p}_j$, so:

$$\frac{\partial M_i\left[\boldsymbol{\alpha}^{\mathrm{b}}\right]}{\partial\boldsymbol{\phi}(t_i)}\mathbf{p}_j \approx \frac{\boldsymbol{\phi}\left\{\tilde{\mathbf{A}}\left[\boldsymbol{\alpha}^{\mathrm{b}}, \tilde{\mathbf{b}} + \epsilon\mathbf{p}_j\right]\right\} - \boldsymbol{\phi}\left\{\tilde{\mathbf{A}}\left[\boldsymbol{\alpha}^{\mathrm{b}}, \tilde{\mathbf{b}}\right]\right\}}{\epsilon}. \tag{30}$$

The size of ϵ is irrelevant for linear system operators, though for nonlinear operators a relative small value ($\epsilon \approx 0.01$) gives more reliable approximations. The dimension of the reduced model is determined by the number of estimations n and the number of basis vectors m in **P**.

5.4. Adjoint Model

[23] An important advantage of the RISM is that an implementation of the adjoint is relatively simple (elaborated in Appendix F) and yields the exact gradient

$$\frac{\Delta\hat{J}}{\Delta\alpha_j} = -\sum_{i=1}\left[\boldsymbol{\lambda}^{\mathrm{r}}(t_i)\right]^{\mathrm{T}} - \begin{bmatrix}\mathbf{M}^{\beta}\\ \mathbf{0}\end{bmatrix}_i \tilde{\mathbf{Z}}^{\mathrm{T}}\mathbf{z}_j\mathrm{d}\alpha_j. \tag{31}$$

Since the reduced model operates with $\Delta\beta_j$s instead of $\Delta\alpha_j$s, it is necessary to project $\mathbf{z}_j\mathrm{d}\alpha_j$ onto $\tilde{\mathbf{z}}$. This seems not logical as we could optimize $\Delta\beta_j$ instead of $\Delta\alpha_j$. The main reason for not doing so is that it distinguishes the method better and makes it directly comparable to the other algorithms that optimize α_js. Moreover, it does not affect the efficiency.

6. Computational Efficiency

[24] The primary factors that determine the efficiency of the RIGM and RISM are quantified by the number of original model simulations τ which is influenced by the snapshots and outer iterations.

6.1. Snapshots

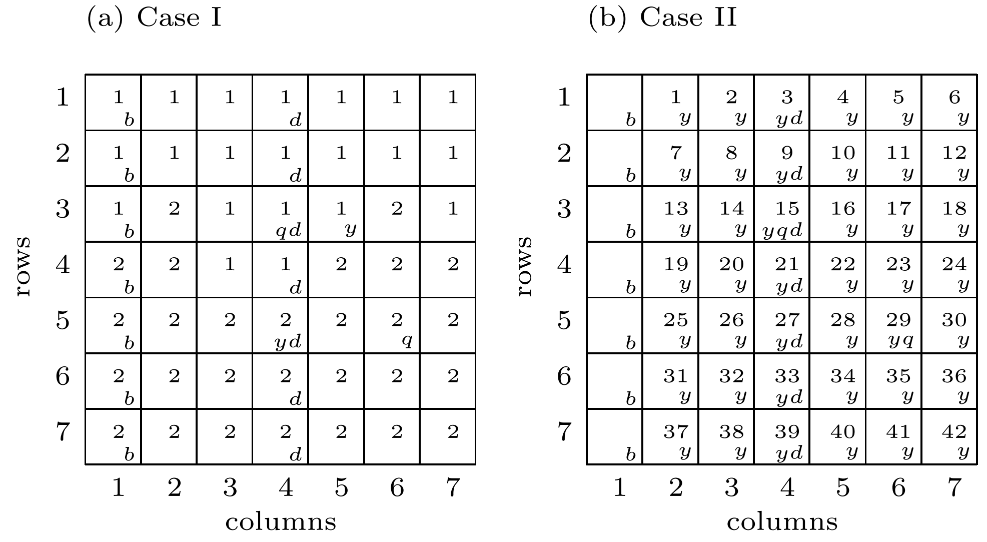
[25] 1. The maximum number of snapshot simulations n is equal to the number of estimation variables times the number of time steps. However, it is difficult to mention the minimum number since there is a tradeoff between the effort put in the snapshot simulations and the "$\hat{J}$ reduction capacity" of the evolved reduced model. For example, it is possible to start with half the maximum number of simulations such that the optimization procedure will focus itself on the major directions first, and add more snapshots as the process proceeds.

[26] 2. More efficiency can be obtained by interchanging $\boldsymbol{\alpha}_i^{\mathrm{s}}$ during the model simulation (dual estimation; see Figure 3b). Unfortunately, this is not possible for the RISM (section 5), since all snapshots are needed for the computation of $\partial M_i/\partial\beta_j$ (24).

[27] 3. Especially, for the RISM it is more time efficient to avoid the time dependency of matrix M_i^{r}. This can easily be done by applying a constant time discretization $\Delta t(t_1, t_n) = c$ and linearizing the nonlinear conductance matrix $\mathbf{C}_{\mathrm{t}}$ for the initial state $\boldsymbol{\phi}^{\mathrm{b}}[\boldsymbol{\alpha}^{\mathrm{b}}](t_0)$. The error that arises will affect the objective function, but is mainly averaged away after projection.

6.2. Outer Iterations

[28] 1. The number of outer iterations is determined by the chosen termination criterion γ (10). It should be chosen not too small ($\gamma \geq 0.1$), as this causes jumping of



(a) Case I

rows	1	2	3	4	5	6	7
1	1 b	1	1	1 d	1	1	1
2	1 b	1	1	1 d	1	1	1
3	1 b	2	1	1 qd	1 y	2	1
4	2 b	2	1	1 d	2	2	2
5	2 b	2	2	2 yd	2	2 q	2
6	2 b	2	2	2 d	2	2	2
7	2 b	2	2	2 d	2	2	2

(b) Case II

rows	1	2	3	4	5	6	7
1	b	1 y	2 y	3 yd	4 y	5 y	6 y
2	b	7 y	8 y	9 yd	10 y	11 y	12 y
3	b	13 y	14 y	15 yqd	16 y	17 y	18 y
4	b	19 y	20 y	21 yd	22 y	23 y	24 y
5	b	25 y	26 y	27 yd	28 y	29 yq	30 y
6	b	31 y	32 y	33 yd	34 y	35 y	36 y
7	b	37 y	38 y	39 yd	40 y	41 y	42 y

Figure 4. Model network for (a) case I and (b) case II with the position of boundary conditions ($b \in \Omega_b$, $d \in \Omega_d$, $q \in \Omega_q$) and observations $y \equiv y^o$. The definition of the zones α_j are given in the grid centers.

suboptimal posterior estimates around the global optimal solution. Since the reduced model probably overestimates α_j^t due to the applied linearizations, it probably yields a redundancy of ζ whenever a threshold ψ_j is applied. It prevents the model from updating the variables to extraordinary values, far beyond background estimation α_j^b that was used for the linearization.

[29] 2. Furthermore, an improved fit between J and $\hat{J}$ can be obtained by applying an appropriate scaling for the estimation variables. Since the estimate variable α_j is log related to the observation, the appropriate scaling here is a log transformation such that $\log(\Delta\alpha_j) \rightarrow \Delta\beta_j$ (23).

7. Examples

7.1. Introduction

[30] This section illustrates the RIGM and RISM by means of twin experiments in which observations are generated from a run of the original model defined to be the "truth." We have used a synthetic 2-D model to generate a set of observations (first experiment). This model consisted of a network of 7 columns by 7 rows with $\Delta x \equiv \Delta y = 10$ m, $\mathbf{T} = 100$ m^2 day^{-1}, and $\mathbf{S} = 0.27$ for the entire model domain. The left side of the model was determined by Dirichlet boundaries ($\phi^b = 0$ m). A drainage element that behaved nonlinearly was active along the entire fourth column for which the flux q_d was computed as

$$q_d(x,\phi) = \begin{cases} \phi(x) < z(x) & ; \quad 0 \\ \phi(x) \geq z(x) & ; \quad C_d[\phi(x) - z_d(x)] \end{cases} \qquad (32)$$

where z_d is a drainage level [L] ($\mathbf{z}_d(t_1, t_n) = -1$ m), and C_d is the drainage conductance ($\mathbf{C}_d = 10$ m^2 day^{-1}). The system was simulated for 10 time steps with $\Delta t(t_1, t_n) = 10$ days.

[31] We have applied RIGM and RISM ($\varphi^e = 99.99\%$) to this problem in order to minimize $\hat{J}$ from an incorrect prior estimate (second experiment). We have aborted the outer iteration cycle ζ when the maximal absolute update of the parameters was $\gamma = 0.1$. For all cases the observational weight matrix $\mathbf{W} = \mathbf{I}$ and we used the Quasi Newton method to proceed the optimization [*Press et al.*, 1992]. We have estimated 2 variables to visualize the optimization progress (case I), and extended this to 42 variables to describe the robustness of the methods (case II).

[32] For purposes of comparison, we have also estimated these problems with the FDM ($\Delta\alpha_j = 0.01$, see (6)) and the ADJ method. Unless stated differently, the number of evaluations of the original model τ is used to compare the different methods. This included the number of evaluations that were eventually needed for the necessary line searches.

7.2. Case I

7.2.1. Configuration

[33] For this particular case, we defined two observation wells $y_1^o(3, 5)$, $y_2^o(5, 4)$ and two pumping wells $q_1(3, 4)$, $q_2(5, 6)$ that varied randomly in strength such that $-2 < q_i(t_1, t_n) < 2$ m day^{-1}. Since no other flux dependencies were active $\mathbf{C}_t = \mathbf{C}_d$ and $\mathbf{z}_t = \mathbf{z}_d$. Two zones were defined for which the prior estimates were $\alpha_1^0 = 0.10$, $\alpha_2^0 = 10.0$ (see Figure 4a).

7.2.2. Results

[34] The ADJ minimized J within $\eta = 9$ gradient iterations and consumed in total $\tau = 30$ original model simulations (see Figure 5a).

[35] The surface of $\hat{J}$, as computed by RIGM (Figure 5b) resembles the surface of J near the locations for which the

Figure 5. Surface of log(J) and the accompanying gradient (arrows) for the optimization of two prior estimates (log(α_1) and log(α_2) presented on the y axis and x axis, respectively) that start at the solid circle and end at the open Circle. (a) ADJ for the original model, (b and c) results from a RIGM optimization, and (d–f) RISM performance. The gray area within Figur es 5b–5f depicts the domain of the reduced model in estimation space (ψ), and the stars represent the chosen combinations of variables within the snapshots.

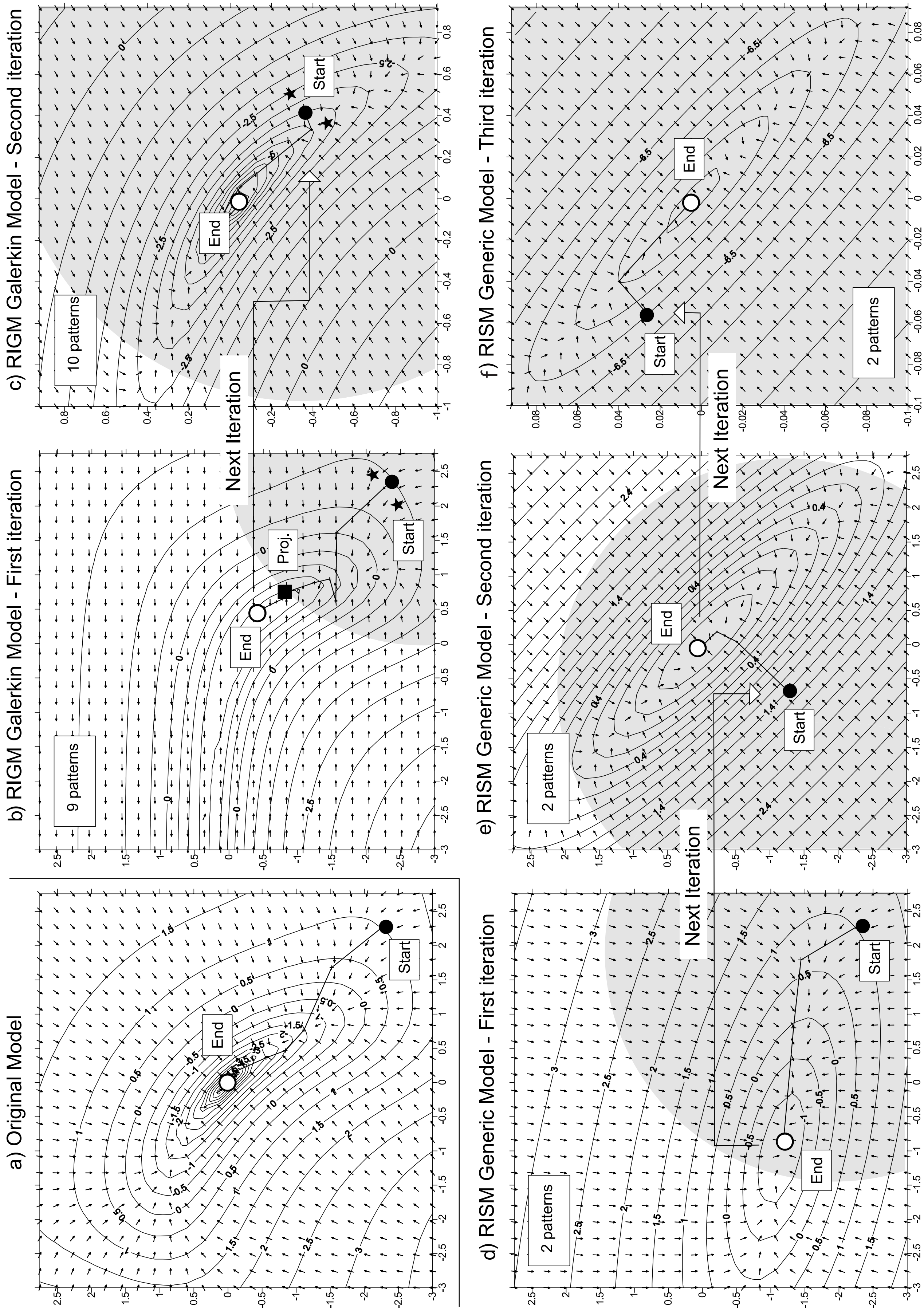

a) Original Model
b) RIGM Galerkin Model - First iteration
c) RIGM Galerkin Model - Second iteration
d) RISM Generic Model - First iteration
e) RISM Generic Model - Second iteration
f) RISM Generic Model - Third iteration
9 patterns
10 patterns
2 patterns
2 patterns
2 patterns
Start
End
Proj.
Next Iteration
Next Iteration
Next Iteration

Figure 5

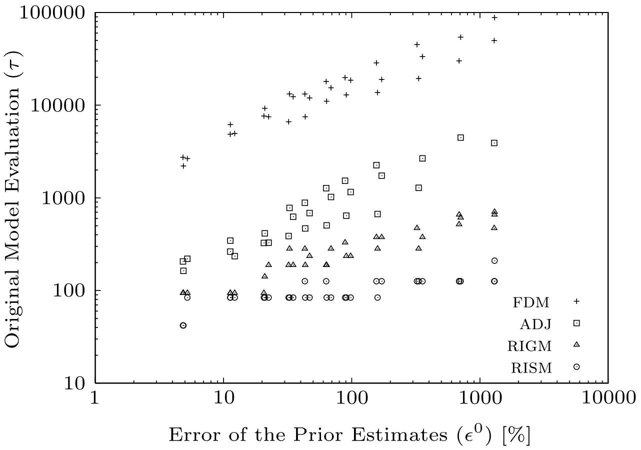


Figure 6. Optimization efficiencies for different methodologies (FDM, ADJ, RIGM, and RISM), expressed in the number of model evaluation τ for 33 samples of 42 incorrect prior estimates.

snapshots were computed (denoted by the black stars). The inaccuracy increases further away since the snapshots and the applied linearization subject to α_j^{b}, are both representative for a limited space (approximated by the grey circle with radius ψ). Nevertheless, a suboptimal minimum was found (after another ξ iteration denoted by the term "Proj." in Figure 5b). An additional second outer iteration ζ was eventually carried out to obtain the closest fit between a priori estimated and the "truth" (see Figures 5b and 5c).

[36] The surface of $\hat{J}$ for the RISM methods looks quite different compared to the others (Figure 5d) since it evolves from a linear model for which α_j was log-transformed prior to the projection. Therefore the log $\hat{J}$ yields a perfect quadratic function for which the innovation of the variables needs to be restricted (grey area with radius ψ) since the model tends to overestimate the variables. %due to the applied linearisation. So, instead of approaching the minimum mainly from one direction, it approaches the "truth" by decreasingly jumping around it from side to side. Hence it needs more outer iterations ζ to find the optimal solution (see Figures 5e and 5f).

[37] Since the RIGM needs to describe the variance on the locations of observation and those for the locations where model forcings exist, it results a 10-dimensional model. This is still significantly less than the original dimensions, but the RISM method needs only two dimensions as it describes the estimation variables subject to the observational variance only. Above all, both methods RIGM and RISM used τ = 4 and τ = 9 original simulations which is less than the original ADJ that consumed τ = 30.

7.3. Case II

7.3.1. Introduction

[38] The efficiency of an inverse modeling problem depends mainly on whether the prior estimates α_j^0 are close to the optimal value α_j^{t}. To obtain insight in the efficiency and robustness of the methods, 33 samples of α_j^0 were generated [*Iman and Shortencarier*, 1984] from which J and $\hat{J}$ were minimized. The quality of these optimizations were determined by the relative error

$$\varepsilon = 100 \cdot \sqrt{\frac{1}{n}\sum_{j=1}^{n}\begin{cases}[\alpha_j]^2 & ; \quad \alpha_j \geq 1.0\\ [\alpha_j]^{-2} & ; \quad 0 < \alpha_j < 1.0,\end{cases}} \qquad (33)$$

since the "truth" is $\|\boldsymbol{\alpha}^{\mathrm{t}}\| = 1.0$. The equivalent ε^0 expresses the error of the prior estimates.

7.3.2. Configuration

[39] The identical model configuration as described for case I (section 7.2) was extended with 42 observations throughout the model (see Figure 4b).

7.3.3. Results

[40] The robustness (i.e., the ability to find the correct posterior estimates) of the methods are mutual comparable for slight errors of the prior estimate (see Figure 6). The RIGM and RISM appear to be more robust for increased values for ϵ^0 since they compute a "rough" surface of the objective function that make them less sensitive to local minima. They are based upon a statistically linearized model that tends to produce a spatially averaged gradient. For all methods τ increases with ϵ^0 and both IGPM and ISPM need less τ compared to the FDM and ADJ methods. The RIGM is approximately two times more efficient than the RISM, since the latter needs additional model evaluations to determine the coefficients of the system dynamics. These results are specific for the synthetic case considered, and will be influenced in practice by a number of topics (e.g., the number of nodes, the number of estimate variables, the number of observations and the complexity of the nonlinearity inside the system). It is reasonable to suggest that all methods will be equally influenced by these factors and that RIGM and RISM retain their efficiencies for other cases.

7.4. Case III

7.4.1. Configuration

[41] This case has a realistic configuration that describes the entire region of the province of Noord-Brabant in the Netherlands ($\approx$10,700 km^2). The model consist of 64 rows, 107 columns and 9 model layers, so the total number of nodes is 61,632. The first model layer contains the influence of an intense surface water network that consists of wetlands and natural dewatering systems situated in higher areas. Absolute levels for these nonlinear boundary conditions were obtained by accurate laser altimetry. The model is further characterized by a detailed description of the precipitation and evapotranspiration rate. These were obtained by combining the rain gauge stations throughout the model domain with accurate land use classification from satellite images.

7.4.2. Results

[42] We have minimized an objective function subject to a change in transmissivity within each model layer (A1). It was based upon 234 synthetic measurements distributed throughout the model and observed for 36 time steps. For the various optimization techniques as described in this paper, the evolution of the objective function value for this minimization is depicted in Figure 7. Although the final objective function value differs slightly between the distinguished methods because of the termination criterium, their valued estimation variables were close to each other. The results found here, for a realistic case, are similar to the

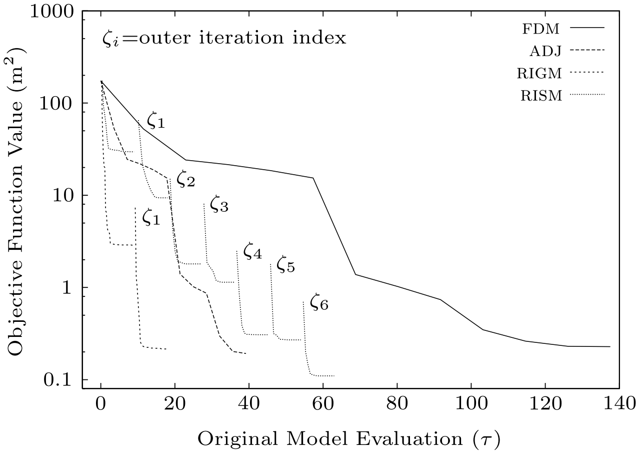


Figure 7. Evolution of the objective function value subject to nine estimation variables for different methodologies (FDM, ADJ, RIGM, and RISM) versus the number of model evaluation τ.

results described for the synthetic cases in the previous subsections. That is: most efficient is the RIGM method, it is almost twice as efficient than the RISM method. The latter is less efficient than the ADJ method. However, it is easy to program and twice as efficient as the FDM method.

8. Conclusions

[43] In this paper two different methodologies have been described for solving nonlinear inverse modeling problems. They were derived by a linear transformation, known as a ''change of coordinates'', that projects a system of equations from a high dimension into a low dimension described by patterns (POD). These PODs were based upon several evaluations of the original model that obtained insight in the observational behavior, subject to the chosen estimation variables. By truncation of unimportant patterns that represented a tiny amount of model behavior, it yielded a low-dimensional model. The first model was based upon a projection of the original PDE (RIGM method) and the second model resulted from a projection of an autoregressive representation of the original system (RISM). Both methods were applied to several synthetic cases and they appeared to be more computationally efficient than the classic inverse methodologies (method of finite differences and the adjoint method). The advantage of the RIGM method is that it conserves the system dynamics in reduced space and is, therefore more accurate from a mathematical point of view. On the other hand this can be disadvantageous too, as it includes some analytical derivatives that are inconvenient to program and nonlinear aspects that cause an extra inner iteration loop. In this perspective the RISM is more flexible and faster since it can handle these nonlinearities more effectively. Above all, both methods operate in less dimensions than the original model and have the ability to become more robust than the classic inverse methodologies, since they focus on the main directions in observational space and tend to become less sensitive to local distortions of the objective function. This should be examined further in future.

[44] Finally, we would like to mention that a reduced model represents always the scale of the original model under consideration. So, a reduced model and the posterior estimates obtained from it, become naturally more accurate to a ''realistic observable situation'' whenever the original model has a denser grid [*Vermeulen et al.*, 2006].

Appendix A: Hydraulic Conductance

[45] In (1) the hydraulic conductance is introduced that is aligned along a direction x_i. For a location $x \in \Omega$ this can be computed as

$$C_i(x) = [\Delta w_i(x) + \Delta w_i(x^*)]\left[\frac{\Delta w_i(x)}{\alpha_j T(x)} + \frac{\Delta w_i(x^*)}{\alpha_j T(x^*)}\right]^{-1}, \qquad \text{(A1)}$$

where Δw_i [L] is the width of the grid cell along the *i*th direction, T [L^{2}T^{-1}] is the transmissivity, and $(\cdot)^*$ denotes the neighbor grid cells along the current direction *i*.

Appendix B: Derivation of the Original Adjoint Model

[46] Equation (7) introduces the adjoint model $\boldsymbol{\lambda}(t_n, t_1)$ that can be concisely elaborated by stating that:

$$\begin{aligned} J = & \sum_{i=1}\left[(\mathbf{y}^{\mathrm{o}} - \mathbf{H}\boldsymbol{\phi})^{\mathrm{T}}\mathbf{W}(\mathbf{y}^{\mathrm{o}} - \mathbf{H}\boldsymbol{\phi})\right](t_i) \\ & + \sum_{i=1}[\boldsymbol{\lambda}(t_i)]^{\mathrm{T}}[\mathbf{A}_i\boldsymbol{\phi}(t_i) - \mathbf{b}(t_i)], \end{aligned} \qquad \text{(B1)}$$

and apply a Taylors series expansion for the variables $\boldsymbol{\phi}(t_i)$, $\boldsymbol{\phi}(t_{i-1})$, α_j, and $\boldsymbol{\lambda}(t_i)$. This yields the expression:

$$\begin{aligned} \Delta J = & \sum_{i=1}[\boldsymbol{\lambda}(t_i)]^{\mathrm{T}}\left\{\mathbf{A}_i\Delta\boldsymbol{\phi}(t_i) - \frac{1}{\Delta t}\mathbf{S}\Delta\boldsymbol{\phi}(t_{i-1})\right. \\ & \left. + \frac{\partial \mathbf{A}}{\partial \alpha_j}\boldsymbol{\phi}(t_i)\Delta\alpha_j\right\} + \frac{\partial J}{\partial \boldsymbol{\phi}(t_i)}\Delta\boldsymbol{\phi}(t_i), \end{aligned} \qquad \text{(B2)}$$

from which a model can be derived that simulates the adjoint state vector backwards in time so:

$$\mathbf{A}_i^{\mathrm{T}}\boldsymbol{\lambda}(t_i) = -\frac{1}{\Delta t}\mathbf{S}\boldsymbol{\lambda}(t_{i+1}) - 2\mathbf{W}[\mathbf{y}^{\mathrm{o}}(t_i) - \mathbf{H}\boldsymbol{\phi}(t_i)]. \qquad \text{(B3)}$$

[47] Eventually, the corresponding gradient becomes:

$$\frac{\Delta J}{\Delta \alpha_j} = \sum_{i=1}[\boldsymbol{\lambda}(t_i)]^{\mathrm{T}}\left(\frac{\partial \mathbf{A}_i}{\partial \alpha_j}\right)\boldsymbol{\phi}(t_i). \qquad \text{(B4)}$$

[48] The time-dependency of $\mathbf{A}_i$ is caused by the nonlinear conductance $\mathbf{C}_{\mathrm{t}}^{\theta}$ (3).

Appendix C: Derivation of the First Derivative of the Hydraulic Conductance

[49] Within the elaboration of the reduced Galerkin model, the partial derivative of the hydraulic conductance

C_i with respect to the estimation variable $\Delta\alpha_j$ is introduced. For each location $x \in \Omega$ and a specific direction i this can be expressed as:

$$\frac{\partial C_i(x)}{\partial \alpha_j} = \frac{\partial C_i(x)}{\partial T^{\mathrm{b}}(x)} \cdot \frac{\partial T^{\mathrm{b}}(x)}{\partial \alpha_j}, \tag{C1}$$

and by using (A1) this yields:

$$\begin{aligned}\frac{\partial C_i(x)}{\partial \alpha_j} = {} & \frac{\Delta w_i(x)[\Delta w_i(x) + \Delta w_i(x^*)]}{[T^{\mathrm{b}}]^2(x)\left[\frac{\Delta w_i(x)}{T^{\mathrm{b}}(x)} + \frac{\Delta w_i(x^*)}{T^{\mathrm{b}}(x^*)}\right]^2} \cdot T^{\mathrm{b}}(x) \\ & + \frac{\Delta w_i(x^*)[\Delta w_i(x) + \Delta w_i(x^*)]}{[T^{\mathrm{b}}]^2(x^*)\left[\frac{\Delta w_i(x)}{T^{\mathrm{b}}(x)} + \frac{\Delta w_i(x^*)}{T^{\mathrm{b}}(x^*)}\right]^2} \cdot T^{\mathrm{b}}(x^*),\end{aligned} \tag{C2}$$

where the background transmissivity is defined as:

$$T^{\mathrm{b}} = \left[\alpha_j^{\mathrm{b}}\right]^{\zeta} T, \tag{C3}$$

that depends on the background estimation α_j^{b} for the current outer iteration cycle ζ.

Appendix D: Derivation of the Reduced Adjoint Model

[50] The formulation of a reduced adjoint model can be concisely derived by a Taylor series expansion (B2) for the variables $\boldsymbol{\lambda}^{\mathrm{r}}$, $\mathbf{r}(t_i)$, $\mathbf{r}(t_{i+1})$ and α_j. It yields eventually a reduced adjoint model:

$$\begin{aligned}&\left[\mathbf{N}^{\mathrm{r}} + \sum_{j=1} \mathbf{U}_j^{\mathrm{r}} \Delta\alpha_j - \frac{1}{\Delta t}\mathbf{S}^{\mathrm{r}} - \mathbf{C}_{\mathrm{t}}^{\mathrm{r}}\right]^{\mathrm{T}} \boldsymbol{\lambda}^{\mathrm{r}}(t_i) \\ &\quad = -\frac{1}{\Delta t}\mathbf{S}^{\mathrm{r}}\boldsymbol{\lambda}^{\mathrm{r}}(t_{i+1}) + \frac{\partial \hat{J}}{\partial \mathbf{r}}(t_i),\end{aligned} \tag{D1}$$

that needs to be solved backwards in time whereby

$$\frac{\partial \hat{J}}{\partial \mathbf{r}} = -2\mathbf{W}^{\mathrm{r}}\left[\mathbf{P}^{\mathrm{T}}\left(\mathbf{y}^{\mathrm{o}} - \mathbf{H}\boldsymbol{\phi}^{\mathrm{b}}\right) - \mathbf{r}\right](t_i), \tag{D2}$$

where the reduced weight matrix $\mathbf{W}^{\mathrm{r}}$ is computed as:

$$\mathbf{W}^{\mathrm{r}} = \mathbf{P}^{\mathrm{T}}\mathbf{W}\mathbf{P}. \tag{D3}$$

[51] Eventually, the gradient can be obtained by applying

$$\frac{\Delta J}{\Delta \alpha_j} = \sum_{i=1} [\boldsymbol{\lambda}^{\mathrm{r}}(t_i)]^{\mathrm{T}} \mathbf{U}_j^{\mathrm{r}} \mathbf{r}(t_i). \tag{D4}$$

Appendix E: Perturbation of the Hydraulic Conductance Along a Pattern Direction

[52] In (24) the system matrix $\tilde{\mathbf{A}}[\boldsymbol{\alpha}_j^{\mathrm{b}} + \Delta\beta_j\tilde{\mathbf{z}}_j]$ is defined and for each location $x \in \Omega$ it can be computed for the elements $C(x)$ in matrix $\mathbf{A} \in \tilde{\mathbf{A}}$ as:

$$\begin{aligned}C_{ij}(x) = {} & [\Delta w_i(x) + \Delta w_i(x^*)] \\ & \cdot \left[\frac{\Delta w_i(x)}{T_j^{\mathrm{b}}(x) + \{\Delta\beta\tilde{z}T^{\mathrm{b}}(x)\}_j} + \frac{\Delta w_i(x^*)}{T_j^{\mathrm{b}}(x^*) + \{\Delta\beta\tilde{z}T^{\mathrm{b}}(x^*)\}_j}\right]^{-1},\end{aligned} \tag{E1}$$

where T_j^{b} is defined in (C3) and the usage of $(\cdot)^*$ is explained in Appendix A.

Appendix F: Derivation of the Reduced Adjoint Model

[53] The reduced adjoint model $\boldsymbol{\lambda}^{\mathrm{r}}(t_n,\ t_1)$ (31) can be concisely elaborated by stating:

$$\begin{aligned}\hat{J} = {} & \sum_{i=1}\left\{\left[\mathbf{P}^{\mathrm{T}}\left(\mathbf{y}^{\mathrm{o}} - \mathbf{H}\boldsymbol{\phi}^{\mathrm{b}}\right) - \mathbf{r}^{\mathrm{p}}\right]^{\mathrm{T}} \mathbf{P}^{\mathrm{T}}\mathbf{W}\mathbf{P}.\right. \\ & \left.\cdot\left[\mathbf{P}^{\mathrm{T}}\left(\mathbf{y}^{\mathrm{o}} - \mathbf{H}\boldsymbol{\phi}^{\mathrm{b}}\right) - \mathbf{r}^{\mathrm{p}}\right]\right\}(t_i) \\ & + \sum_{i=1} [\boldsymbol{\lambda}^{\mathrm{r}}(t_i)]^{\mathrm{T}}[\mathbf{r}^{\mathrm{p}}(t_i) - \mathbf{M}_i^{\mathrm{p}}\mathbf{r}^{\mathrm{p}}(t_{i-1})],\end{aligned} \tag{F1}$$

with the partitioned matrices $(\cdot)^{\mathrm{p}}$ defined as:

$$\mathbf{M}_i^{\mathrm{p}} = \begin{bmatrix} \mathbf{M}^{\mathrm{r}} & \mathbf{M}^{\beta} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}_i, \tag{F2}$$

$$\mathbf{r}^{\mathrm{p}}(t_i) = \begin{bmatrix} \mathbf{r} \\ \Delta\boldsymbol{\beta} \end{bmatrix}(t_i). \tag{F3}$$

[54] After a Taylor series expansion (B2) for the variables, $\boldsymbol{\lambda}^{\mathrm{r}}(t_i)$, $\mathbf{r}^{\mathrm{p}}(t_i)$, $\mathbf{r}^{\mathrm{p}}(t_{i-1})$ and $\Delta\boldsymbol{\alpha}_j$, the expression for the reduced adjoint model yields:

$$\boldsymbol{\lambda}^{\mathrm{r}}(t_{i-1}) = [\mathbf{M}_i^{\mathrm{p}}]^{\mathrm{T}}\boldsymbol{\lambda}^{\mathrm{r}}(t_i) + \mathbf{e}(t_{i-1}), \tag{F4}$$

where

$$\mathbf{e}(t_{i-1}) = \begin{bmatrix} -2\mathbf{W}^{\mathrm{r}}\left\{\mathbf{P}^{\mathrm{T}}\left(\mathbf{y}^{\mathrm{o}} - \mathbf{H}\boldsymbol{\phi}^{\mathrm{b}}\right) - \mathbf{r}\right\} \\ 0 \end{bmatrix}(t_{i-1}), \tag{F5}$$

and the reduced observational weights $\mathbf{W}^{\mathrm{r}}$ are computed by D3. The gradient eventually becomes

$$\frac{\Delta \hat{J}}{\Delta \alpha_j} = \sum_{i=1} [\boldsymbol{\lambda}^{\mathrm{r}}(t_i)]^{\mathrm{T}} \frac{\partial \mathbf{r}^{\mathrm{p}}(t_i)}{\partial \Delta\alpha_j}, \tag{F6}$$

where

$$\frac{\mathbf{r}^{\mathrm{p}}(t_i)}{\partial \Delta\alpha_j} = \frac{\partial \mathbf{r}(t_i)}{\partial \beta_j} \cdot \frac{\partial \beta_j}{\partial \alpha_j}, \tag{F7}$$

$$\frac{\mathbf{r}^{\mathrm{p}}(t_i)}{\partial \Delta\beta_j} = -\begin{bmatrix} \mathbf{M}^{\beta} \\ \mathbf{0} \end{bmatrix}_i, \text{ and} \tag{F8}$$

$$\frac{\partial \beta_j}{\partial \alpha_j} = \tilde{\mathbf{Z}}^{\mathrm{T}}\mathbf{z}_j \mathrm{d}\alpha_j. \tag{F9}$$

[55] In the situation where $\boldsymbol{\alpha}_j$ is log transformed the expression remains identical since $\mathrm{d}[\log(\alpha_j)] = 1/\alpha_j$ for which $\alpha_j = 1.0$.

References

Atwell, J. A., and B. B. King (2004), Reduced order controlleres for spatially distributed systems via proper orthogonal decomposition, *SIAM J. Sci. Comput.*, *26*(1), 128–151.

Bennett, A. F. (1992), *Inverse Methods in Physical Oceanography*, Cambridge Univ. Press, New York.

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.

Courtier, P. (1997), Dual formulation of four-dimensional variational assimilation, *Q. J. R. Meteorol. Soc.*, *123*, 2449–2461.

Courtier, P., J. N. Thepaut, and A. Hollingsworth (1994), A strategy for operational implementation of 4D-Var, using an incremental approach, *Q. J. R. Meteorol. Soc.*, *120*, 1367–1387.

Cooley, R. L. (1985), A comparison of several methods of solving nonlinear regression groundwater flow problems, *Water Resour. Res.*, *21*(10), 1525–1538.

Courant, R., and D. Hilbert (1953), *Methods of Mathematical Physics*, Wiley-Interscience, Hoboken, N. J.

Edwards, M. G. (2000), M-matrix flux splitting for general full tensor discetization operators on structered and unstructered grids, *J. Comput. Phys.*, *160*, 1–28.

Hammond, G. E., A. J. Valocchi, and P. C. Lichtner (2005), Application of Jacobian-free Newton-Krylov with physics-based preconditioning to biogeochemical transport, *Adv. Water Resour.*, *28*, 359–376.

Hoffmann Jørgensen, B., and J. N. Sørensen (2000), Proper orthogonal decomposition and low-dimensional modelling, *ERCOFTAC Bull.*, *46*, 44–51.

Holmes, P., J. L. Lumley, and G. Berkooz (1996), *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*, Cambridge Univ. Press, New York.

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.

Lawless, A. S., S. Gratton, and N. K. Nichols (2005), Approximate iterative methods for variational data assimilation, *Int. J. Numer. Methods Fluids*, *47*, 1129–1135.

Lee, K. S., E. Yongtae, J. W. Chung, J. Choi, and D. Yang (2000), A control-relevant model reduction technique for lonlinear systems, *Comput. Chem. Eng.*, *24*, 309–315.

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*.

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*, 33 pp.

Mehl, S. W., and M. C. Hil (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., Univ. of Md., College Park.

Newman, A. J., and P. S. Krishnaprasad (1998), Nonlinear model reduction for RTCVD, *Tech. Rep. T. R.98-10*, Inst. for Syst. Res., Univ. of Md., College Park.

Park, H. M., and C. H. Cho (1996), Low dimensional modeling of flow reactors, *Int. J. Heat Mass Transfer*, *39*(16), 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.

Pearson, K. (1901), On lines and planes of closest fit to points in space, *Philos. Mag.*, *2*, 609–629.

Poeter, E. P., and M. C. Hill (1997), Inverse methods: A necessary next step in groundwater modeling, *Ground Water*, *35*(2), 250–260.

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.

Sirovich, L. (1987), Turbulence and the dynamics of coherent structures; part I: Coherent structures, *Q. Appl. Math.*, *45*(3), 561–571.

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*(12), 1851–1860.

van Doren, J., R. Markovinović, and J. D. Jansen (2005), Reduced-order optimal control of water flooding using proper orthogonal decomposition, *Comput. Geosci.*, *10*, 137–158, doi:10.1007/s10596-005-9014-2.

Vermeulen, P. T. M., and A. W. Heemink (2005), Model-reduced variational data assimilation, *Mon. Weather Rev.*, in press.

Vermeulen, P. T. M., A. W. Heemink, and C. B. M. te Stroet (2004a), Low-dimensional modeling of numerical groundwater flow, *Hydrol. Processes*, *18*, 1487–1504.

Vermeulen, P. T. M., A. W. Heemink, and C. B. M. te Stroet (2004b), Reduced models for linear groundwater flow models using empirical orthogonal functions, *Adv. Water Resour.*, *27*(1), 57–69.

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

Vermeulen, P. T. M., A. W. Heemink, and C. B. M. te Stroet (2006), Limitations to upscaling of groundwater flow models dominated by surface water interaction, *Water Resour. Res.*, doi:10.1029/2005WR004620, in press.

Wen, X. H., L. J. Durlofsky, and M. G. Edwards (2003), Use of border regions for improved permeability upscaling, *Math. Geol.*, *35*, 521–547.

Winter, C. L., E. P. Springer, K. Costigan, P. Fasel, S. Mniszewski, and G. Zyvoloski (2004), Virtual watersheds: Simulating the water balance of the Rio Grande basin, *Comput. Sci. Eng.*, *6*, 18–26.

Wu, Y. S., K. Zhang, C. Ding, K. Pruess, E. Elmroth, and G. S. Bodvarsson (2002), An efficient parallel-computing method for modeling nonisothermal multiphase flow and multicomponent transport in porous and fractured media, *Adv. Water Resour.*, *25*, 243–261.

A. W. Heemink, Department of Applied Mathematical Analysis, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Mekelweg 4, P.O. Box 5031, 2600 GA Delft, Netherlands.

C. B. M. te Stroet and P. T. M. Vermeulen, TNO Build Environment and Geoscience, Netherlands Geological Survey, P.O. Box 80015, 3508 TA Utrecht, Netherlands. (peter.vermeulen@tno.nl)