Exploiting Sparsity in Solving PDE-Constrained Inverse Problems: Application to Subsurface Flow Model Calibration

Azarang Golmohammadi, M-Reza M. Khaninezhad, and Behnam Jafarpour

Abstract Inverse problems are frequently encountered in many areas of science and engineering where observations are used to estimate the parameters of a system. In several practical applications, the dynamic processes that take place in a physical system are described using a set of partial differential equations (PDEs), which are typically nonlinear and coupled. The inverse problems that arise in those systems ought to be constrained to honour the governing PDEs. In this chapter, we consider high-dimensional PDE-constrained inverse problems in which, because of spatial patterns and correlations in the distribution of physical properties of a system, the underlying parameters tend to reside in (usually unknown) low-dimensional manifolds, thus have sparse (low-rank) representations. The sparsity of the parameters is amenable to an effective and flexible regularization form that can be exploited to improve the solution of such inverse problems. In applications where prior training data are available, sparse manifold learning methods can be adopted to tailor parameter representations to the specific requirements of the prior data. However, a major risk in employing prior training data is the significant uncertainty about the underlying conceptual models and assumptions used to develop the prior. A group-sparsity formulation is discussed for addressing the uncertainty in the prior training data when multiple distinct, but plausible, prior scenarios are encountered. Examples from geosciences application are presented where images of rock material properties are reconstructed from limited nonlinear fluid flow measurements.

A. Golmohammadi · M.-R. M. Khaninezhad
Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA, USA
e-mail: agolmoha@usc.edu; m.khaninezhad@usc.edu

B. Jafarpour (✉)
Mork Family Department of Chemical Engineering and Material Science, University of Southern California, Los Angeles, CA, USA

Ming Hsieh Department of Electrical Engineering, University of Southern California, Los Angeles, CA, USA
e-mail: behnam.jafarpour@usc.edu

H. Antil et al. (eds.), *Frontiers in PDE-Constrained Optimization*, The IMA Volumes in Mathematics and its Applications 163,
https://doi.org/10.1007/978-1-4939-8636-1_12

1 Introduction

The spatiotemporal evolution of dynamic state variables in many physical systems is governed by coupled partial differential equations (PDEs) that are typically derived from the balance laws of physics (mass, momentum, and energy conservation). The observable responses of these dynamical systems can usually be described as a function of their state variables, which in turn depend on model inputs, including controls, initial/boundary conditions, and parameters. In general, the functional relation between model input parameters and observable responses can be expressed as a (typically nonlinear) mapping that involves the solution of the underlying PDEs. Examples of these physical systems include fluid flow and heat transfer processes [64], electromagnetic systems [65], motion of planets in solar system [60], human's neural mechanism [42]. The exponential increase in computing power has enabled considerable advances in numerical simulation of complex processes in large-scale physical systems that have high-dimensional PDEs as governing equations. Advances in computing power have also led to development of computationally demanding inverse modelling algorithms with potentially thousands of forward model simulations, which was once considered infeasible.

The parameters that appear in the governing PDEs of physical systems are either directly observable or they need to be inferred from indirect and often limited observable quantities (outputs) of the system [52, 62, 79, 85, 89]. In some cases, a spatially distributed physical property may only be directly observable at finite points in space, requiring spatial interpolation techniques to predict unobserved parameter values. In general, estimation of model parameters from limited output measurements of the system leads to an inference or inverse problem [59, 78]. In many cases, the inverse modelling formulations involve a minimization problem where the objective function represents the mismatch between model predicted and observed data as well as other terms that penalize departure from prior (explicit or implicit) knowledge about the solution. When the system outputs depend on the solution of the PDEs that establish physical laws (e.g. mass/momentum/energy balance), the resulting inverse problem formulation must ensure that the PDE constraints are honoured, thus leading to a PDE-constrained inverse problem. Including the PDE constraints ensures that the solution of the resulting inverse problem honours the underlying governing equations (i.e. well-established physical laws such as mass/momentum conservation).

Inverse problems that arise in many practical applications are ill-posed, as the measured data are not sufficient to find a unique solution [35, 63]. When there are fewer measurements than unknown model parameters in a system, a situation that is commonly encountered in practice, the problem is underdetermined and cannot have a unique solution. Additional (a priori) information are needed to constrain the solution and eliminate implausible outcomes. A common approach to address solution non-uniqueness is to adopt a probabilistic (Bayesian) inverse modelling framework [1, 26, 32, 51, 53, 78], where the elements of the inverse problem (parameters, data, and forward model) are represented with their respective

uncertainties, typically using probability density functions (PDFs). In this chapter, we focus on deterministic inverse problems. First, an overview of inverse modelling formulation is presented, followed by general strategies for solving ill-posed inverse problems that are constrained by complex PDEs. In numerical solution techniques, the PDEs are solved by first discretizing the domain and assigning input parameters to the discrete cells. This approach leads to a discrete ill-posed inverse problem in which vector representations (as opposed to continuous functions) are used to describe the unknown parameters. The main focus of this chapter is on formulation and solution of such discrete inverse problems in which the parameters are either inherently sparse or can have a sparse approximation.

2 Inverse Problem Formulation

To formulate a general inverse problem, consider collecting the observations of a physical system in a vector $\mathbf{d}$. These observations are related to the parameters of the system through a (generally nonlinear) mapping, i.e. $\mathbf{d} = \mathbf{g}(\mathbf{u})$. Here, $\mathbf{u}$ contains the parameters of the system, and $\mathbf{g}(.)$ is the nonlinear function that maps the parameter space onto the observation space. We assume that the observations $\mathbf{d}$ and the parameters $\mathbf{u}$ are vectors in $\mathbb{R}^{m\times 1}$ and $\mathbb{R}^{n\times 1}$, respectively.

Definition (General Inverse Problem) Consider the *Banach* spaces $\mathscr{U}$ and $\mathscr{D}$, and a mapping $\mathbf{G} : \mathscr{U} \rightarrow \mathscr{D}$. The inverse problem consists of the solution to the equation [66]:

$$\mathbf{g}(\mathbf{u}) = \mathbf{d} \qquad \mathbf{u} \in \mathscr{U} \quad \& \quad \mathbf{d} \in \mathscr{D} \tag{1}$$

If an exact solution is not expected (e.g. due to observation errors), the inverse problem in (1) is expressed as a minimization of the form:

$$\min_{\mathbf{u}} \quad J(\mathbf{u}) = \|\mathbf{g}(\mathbf{u}) - \mathbf{d}\|_2^2 \qquad \mathbf{u} \in \mathscr{U} \tag{2}$$

When the *Banach* space $\mathscr{D}$ is some ℓ^2-space, then this becomes a classical least-squares problem [57].

The simplest form of an inverse problem is obtained when observations and model parameters are related linearly [13, 78], i.e. $\mathbf{d} = \mathbf{G}\mathbf{u} + \boldsymbol{\epsilon}$. Here, $\mathbf{u}$ is the parameter of interest, $\mathbf{G}$ is the linear mapping from parameter space to the observation space, and $\boldsymbol{\epsilon}$ is the observation noise, which is usually considered to be independent of the parameters $\mathbf{u}$. In the linear case, the inverse problem in Equation (2) is expressed as:

$$\min_{\mathbf{u}} \quad \|\mathbf{G}\mathbf{u} - \mathbf{d}\|_2^2 \qquad \text{s.t.,} \qquad \mathbf{u} \in \mathscr{U} \tag{3}$$

with a simple quadratic objective function. In practical applications, when data is noisy, the least-square term in Equation (3) is generalized to $\|\mathbf{C}_{\boldsymbol{\epsilon}}^{-\frac{1}{2}}(\mathbf{Gu}-\mathbf{d})\|_2^2$, where $\mathbf{C}_{\boldsymbol{\epsilon}}$ is the (usually diagonal) noise covariance matrix $\boldsymbol{\epsilon}$. For ill-posed linear inverse problems, the formulation often takes the form:

$$\min_{\mathbf{u}} \quad J(\mathbf{u}) \quad \text{s.t.,} \quad \|\mathbf{d}-\mathbf{Gu}\|_2^2 \leq \sigma^2 \tag{4a}$$

$$\min_{\mathbf{u}} \quad J(\mathbf{u}) + \frac{1}{\lambda^2}(\|\mathbf{d}-\mathbf{Gu}\|_2^2 - \sigma^2) \tag{4b}$$

$$\min_{\mathbf{u}} \quad \|\mathbf{d}-\mathbf{Gu}\|_2^2 + \lambda^2 J(\mathbf{u}) \tag{4c}$$

In Equation (4a), the constraint, i.e. $\|\mathbf{d}-\mathbf{Gu}\|_2^2 \leq \sigma^2$, is added to the objective function by the penalty method [8], and the resulting equation in (4b) is reshaped into Equation (4c) by multiplying the objective function by λ^2. In Equation (4), $J(\mathbf{u})$ is a function that restricts (regularizes) the behaviour/structure of $\mathbf{u}$, and σ^2 is a bound on the observation error. For example, if $\mathbf{u}_0$ is a prior belief about the parameter $\mathbf{u}$, minimization of $J(\mathbf{u}) = \|\mathbf{u}-\mathbf{u}_0\|_2^2$ results in a solution with minimum departure from $\mathbf{u}_0$ [78]. A classical example of regularization functions are the Tikhonov regularization forms [81], for which $J(\mathbf{u})$ is defined as the second norm of the first or second derivatives of the parameters (to promote solution smoothness or flatness, respectively). It is important to note that the regularization parameter λ has a significant impact on the solution by balancing the importance of data misfit and regularization terms. For linear problems, cross validation [31] and L-curve [34] methods have been proposed for finding an optimal value for the regularization parameter.

In many practical problems, the relationship between the observed data and model parameters is nonlinear, i.e. $\mathbf{d} = \mathbf{g}(\mathbf{u}) + \boldsymbol{\epsilon}$ [74, 79]. The corresponding nonlinear inverse problem can be expressed as:

$$\min_{\mathbf{u}} \quad J(\mathbf{u}) \quad \text{s.t.,} \quad \|\mathbf{d}-\mathbf{g}(\mathbf{u})\|_2^2 \leq \sigma^2 \tag{5a}$$

$$\min_{\mathbf{u}} \quad \|\mathbf{d}-\mathbf{g}(\mathbf{u})\|_2^2 + \lambda^2 J(\mathbf{u}) \tag{5b}$$

For physical systems in which the evolution of the state variables is determined by solving PDE systems, the resulting inverse problems include the PDEs as constraints, that is,

$$\min_{\mathbf{u}} \quad \|\mathbf{d}-\mathbf{g}(\mathbf{u})\|_2^2 + \lambda^2 J(\mathbf{u}) \qquad \text{s.t.,} \qquad f(\mathbf{u}, \mathbf{x}(\mathbf{u})) = 0 \tag{6}$$

where $f(\mathbf{u}, \mathbf{x}(\mathbf{u})) = 0$ represents the PDE system. We note that the measurement operator $\mathbf{g}(\mathbf{u})$ is usually a function of the state vector $\mathbf{x}(\mathbf{u})$, which, for compactness, is not explicitly expressed in Equation (6). It is common to enforce the constraints by first solving the PDE system to obtain the state variables and then using them

to predict the measurements. In other words, the PDE system is solved to derive the nonlinear measurements, resulting in predicted measurements that honour the constraints.

In practice, nonlinear inverse problems do not lend themselves to analytical solutions, and iterative numerical optimization techniques must be employed to find the solution. In iterative solution schemes, given the current iterate $\mathbf{u}^{(k)}$, an updated solution is sought by expanding the nonlinear function $\mathbf{g}(\mathbf{u})$ around the current iterate, using either first- or second-order Taylor expansions. For example, when a linear approximation is used, the resulting objective function takes the form:

$$\mathbf{u}^{(k+1)} = \underset{\mathbf{u}}{\operatorname{argmin}} \quad \|\mathbf{d} - (\mathbf{g}(\mathbf{u}^{(k)}) + \mathbf{G}_{\mathbf{u}}(\mathbf{u} - \mathbf{u}^{(k)}))\|_2^2 + \lambda^2 J(\mathbf{u}) \tag{7}$$

where $\mathbf{G}_{\mathbf{u}}$ is the Jacobian matrix that contains the first-order derivative of multivariate vector function $\mathbf{g}(\mathbf{u})$ with respect to entries of $\mathbf{u} = \mathbf{u}^{(k)}$. The linear objective function in Equation (7) can be readily minimized to find $\mathbf{u}^{(k+1)}$, and the process is continued until the algorithm converges to a solution [78].

3 Parametrization and Regularization Techniques

Inverse problems often involve high-dimensional parameters with complex relations that need to be estimated from low-resolution nonlinear data. In addition to numerical stability issues (due to high-dimensional and low-rank nature of the matrices involved) in solving such ill-posed inverse problems, several non-unique solutions can be found that reproduce the (limited) available data, but fail to predict the future response of the system. In some physical systems, the parameters may represent spatially distributed material properties with specific architecture or patterns. In such cases, in addition to dealing with high parameter dimensionality, it is important to preserve the expected spatial structure of the parameters [6, 12, 18, 37, 47, 90]. Parametrization and regularization are two common approaches that aim to achieve these two goals by reducing parameter dimensionality and imparting pre-specified attributes on the solution. Techniques for regularizing the solution of ill-posed inverse problems have been extensively studied in the literature (e.g. see [24, 78, 81, 86]). Regularization is usually implemented by minimizing a penalty function (e.g. $J(\mathbf{u})$ in Equations (4)–(7)) that promotes an attribute of interest in the solution, e.g. using a roughness penalty function to obtain smooth solutions. By imposing certain patterns/attributes on the solution, regularization creates correlation structures that, in effect, implicitly reduce the dimension of the parameter space.

Inverse problem formulations are directly influenced by the choice of parameters (i.e. parameterization or re-parameterization) [24, 39]. Parameterization refers to changing the original parameters of an inverse problem to a (typically much smaller) set of new parameters that facilitate the search for a solution. It is often

used to explicitly reduce the number of unknown parameters, while capturing their main characteristics, with the purpose of alleviating problem ill-posedness. Parameterization can also provide more compact descriptions of complex parameter structures and facilitate their reconstruction. In solving inverse problems, choosing an appropriate domain that affords an effective description of the parameters is complicated by the lack of complete knowledge about the solution. However, a reasonable choice for the parameter domain may be deduced from the knowledge about the physics of the system under analysis and/or based on the past experience. Parameterization can be performed either in the original domain (space/time), in which the PDEs are solved, or they can be implemented by transforming the parameters into a different (often abstract) domain with certain desirable properties.

A linear parameterization [87] can generally be expressed as:

$$\mathbf{u} = \mathbf{\Phi}\mathbf{v} = \sum_{i=1}^{k} \boldsymbol{\phi}_i v_i \tag{8}$$

where $\mathbf{u}$ and $\mathbf{v}$ are vectors of the original and transformed model parameters, respectively; and $\mathbf{\Phi}$ is the linear transformation matrix with columns corresponding to the basis functions (i.e. $\phi_{i:i=1,\ldots,k}$), which are linearly combined, using the entries of $\mathbf{v}$ as coefficients, to yield $\mathbf{u}$. Matrix $\mathbf{\Phi}$ can be viewed as a linear mapping of the transformed parameters $\mathbf{v}$ onto the original parameters $\mathbf{u}$. Different choices of $\mathbf{\Phi}$ lead to alternative parameterization bases (domains) with distinct properties that can be exploited in formulating the inverse problem.

Using the linear relation $\mathbf{u} = \mathbf{\Phi}\mathbf{v}$, it is straightforward to rewrite the inverse problem objective function in Equation (7) in terms of $\mathbf{v}$ as follows:

$$\mathbf{v}^{(k+1)} = \underset{\mathbf{v}}{\operatorname{argmin}} \quad \|\mathbf{d} - (\mathbf{g}(\mathbf{v}^{(k)}) + \mathbf{G}_{\mathbf{v}}(\mathbf{v} - \mathbf{v}^{(k)}))\|_2^2 + \lambda^2 J(\mathbf{v}) \tag{9}$$

where $J(\mathbf{v})$ defines a regularization constraint on the new parameters $\mathbf{v}$ in the transform domain (more details in subsequent sections). Note that the transformation matrix $\mathbf{\Phi}$ is assumed to be constant and dropped for brevity. Furthermore, $\mathbf{G}_{\mathbf{v}}$ in Equation (9) presents the Jacobian matrix of the observations with respect to the transformed coefficients and can be simply calculated through the chain rule of differentiation as:

$$\mathbf{G}_{\mathbf{v}} = \frac{\partial}{\partial \mathbf{v}} \mathbf{g}(\mathbf{v})|_{\mathbf{v}=\mathbf{v}^k} = \mathbf{G}_{\mathbf{u}}\mathbf{\Phi} \tag{10}$$

A nonlinear parameterization can be expressed as $\mathbf{u} = \phi(\mathbf{v})$, where the mapping $\phi(.)$ represents a general nonlinear transformation. For instance, kernel functions provide mappings that can be used to reduce parameter nonlinearity prior to applying a linear parameterization [70, 71, 84]. Kernel-based methods use kernel functions to operate in high-dimensional feature spaces without computing the coordinates of the feature space. Instead, they compute the inner products of

(training) data pairs in the feature space. Using this approach, the inner product of the vectors in the nonlinear space is calculated by kernel functions, $k(\mathbf{v}, \mathbf{v}^{'}) =< \phi(\mathbf{v}), \phi(\mathbf{v}^{'}) >$, where $\phi(.)$ is a feature map (e.g. a polynomial). The kernel $k(\mathbf{v}, \mathbf{v}^{'})$ is a function of $\mathbf{v}$ and $\mathbf{v}^{'}$, and it eliminates the need for nonlinear expansion of the parameters. A major difficulty that arises in implementing nonlinear transformations is the lack of unique back transformation due to the nonlinear form of the transform function $\phi(.)$. In this chapter, linear transforms are discussed.

3.1 Parameterization/Regularization in Space

Zonation Zonation [37] is the simplest spatial parameterization technique in which subsets of the parameter vector $\mathbf{u}$ are assumed to have (approximately) identical values and can be aggregated into a single parameter. In imaging applications where $\mathbf{u}$ is a spatial image (of an unknown property distribution), subsets of entries of $\mathbf{u}$ that correspond to a local neighbourhood in the image form a segment or a zone with identical parameter values. By aggregating such multiple entries into a single parameter, zonation can significantly reduce the number of parameters. Figure 1(a) depicts a sample parameter distribution (shown in x-y plane) that consists of k regions or zones ($\boldsymbol{R}_1, \ldots, \boldsymbol{R}_k$). If the parameter values in each region are similar,

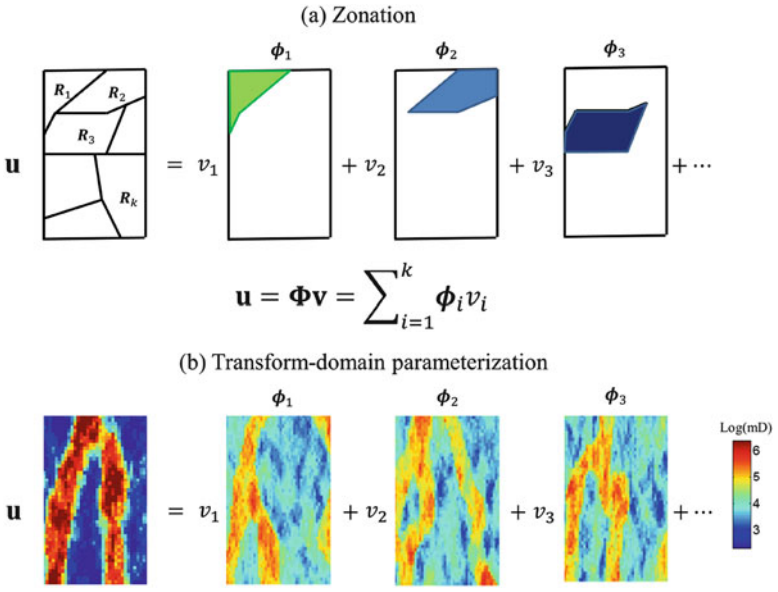

Fig. 1 Schematic of parameter representation via linear expansion: (a) spatial zonation with predefined regions with similar parameter values; (b) expansion with functions derived from compressive transform

the number of parameters can be reduced to $k << n$. This parameterization can be effectively expressed using a general linear expansion representation, consisting of basis vectors $\boldsymbol{\phi}_{s:1\leq s\leq k}$ in which only the entries corresponding to region $\boldsymbol{R}_s$ are non-zero (ones) and the remaining entries are zero (see Figure 1(a)). Using zonation, the formulation of the inverse problem is reduced to:

$$\min_{\mathbf{v}} \quad \|\mathbf{d}-\mathbf{g}(\mathbf{u})\|_2^2+\lambda^2 J(\mathbf{u}) \qquad \text{s.t.,} \qquad \mathbf{u}=\sum_{i=1}^{k}\boldsymbol{\phi}_i v_i \tag{11a}$$

where with the new parameters, i.e. $[v_1 v_2 \dots v_k]$, the problem is better posed (only k unknowns). In many cases, zonation leads to very few zones, eliminating the need for the regularization term, i.e. $J(\mathbf{u})$ in Equation (11a). Therefore, a simpler version of the problem can be expressed as:

$$\min_{\mathbf{v}} \quad \|\mathbf{d}-\mathbf{g}(\mathbf{u})\|_2^2 \qquad \text{s.t.,} \qquad \mathbf{u}=\sum_{i=1}^{k}\boldsymbol{\phi}_i v_i \tag{11b}$$

Although zonation is a simple and intuitive parameterization approach, it suffers from a number of shortcomings. First, it is not trivial to define the zones for an unknown map a-priori. Adaptive multi-resolution zonation techniques [33] have been developed that allow the zones to be redefined (updated) during inversion. Second, the sharp boundaries that separate the zones may not be realistic or plausible. Finally, eliminating the variability (heterogeneity) within each region can result in unintended elimination of local, but important, features and introduce undesired solution bias. Several other parameterization methods have been developed to improve the ill-posedness of inverse problems. Examples of these methods include transform-domain techniques such as the principal component analysis (PCA), the Fourier-based discrete cosine transform (DCT), and the discrete wavelet transform (DWT) (Section 3.2).

Tikhonov Regularization Tikhonov regularization [81] is achieved by minimizing the zeroth-/first-/second-order derivative of the solution to promote minimum-length/smooth/flat solutions, respectively. Tikhonov regularization has been widely applied to inverse problems in several imaging applications, where the parameters are expected to show some degree of continuity. The reason for this attribute is that images that represent the parameters are often related to physical properties that naturally follow certain continuity in their formation. To illustrate how Tikhonov regularization works, consider the local operator that approximates the first-order directional derivative for entry $u_{i,j}$ of the parameter vector $\mathbf{u}$ (defined on a two-dimensional x-y coordinates), i.e. $(\nabla\mathbf{u})_{i,j} \approx \begin{bmatrix} u_{i,j}-\frac{1}{2}(u_{i-1,j}+u_{i+1,j}) \\ u_{i,j}-\frac{1}{2}(u_{i,j-1}+u_{i,j+1}) \end{bmatrix}$. This notation is used to demonstrate the central point finite difference approximation to the first-order directional derivative. Minimizing $\int \|\nabla\mathbf{u}\|_2^2 \mathrm{d}\mathbf{u} \approx \Delta \times$

$\sum_{i,j} \|(\nabla \mathbf{u})_{i,j}\|_2^2$, where Δ denotes a small spatial perturbation, corresponds to solutions that exhibit smooth transition (in parameter values) from $u_{i,j}$ to its neighbouring grid cells. With the first-order Tikhonov regularization, the inverse problem objective function takes the form:

$$\min_{\mathbf{u}} \quad \|\mathbf{d} - \mathbf{g}(\mathbf{u})\|_2^2 + \lambda^2 \int \|\nabla \mathbf{u}\|_2^2 \mathrm{d}\mathbf{u} \tag{12}$$

For discrete problems, the spatial derivatives and the regularization function can be written as a linear operator $\mathbf{W}$; that is, the regularization term can be simplified to $\int \|\nabla \mathbf{u}\|_2^2 \mathrm{d}\mathbf{u} = \|\mathbf{W}\mathbf{u}\|_2^2$.

Total Variation Total variation [27, 50, 69] is a regularization technique that is used to promote piecewise smooth solutions. Hence, the regularization penalty is lenient to solutions that are generally smooth but can have discontinuity in certain parts. This form of regularization is implemented by applying a milder penalty to spatial derivatives of the parameters. In Total Variation, the ℓ_1-norm (instead of the ℓ_2-norm) of the first-order derivative of the solution is minimized. The ℓ_1-norm is less sensitive to larger entries and tends to tolerate discontinuity, which is often exhibited through large directional derivatives. In implementing the total variation, one seeks to minimize the following regularized least-squares form:

$$\min_{\mathbf{u}} \quad \|\mathbf{d} - \mathbf{g}(\mathbf{u})\|_2^2 + \lambda^2 \int \sqrt{\sum_j (\nabla_j \mathbf{u})^2} \mathrm{d}\mathbf{u} \tag{13}$$

where the index j is the number of directional derivatives, and $\nabla_j \mathbf{u}$ calculates the derivative of $\mathbf{u}$ at a direction specified by index j. The total variation regularization can be implemented for any specified direction. In its standard implementation, the directions j are the three Cartesian coordinates.

3.2 Transform-Domain Parameter Representations

Compressive transforms are used to compactly represent/approximate the most salient features of images and signals. In inverse problems, it may be possible to apply a transformation to the original parameters to achieve an effective low-rank representation. Examples of transform-domain representation techniques are those that are used in image compression, e.g. Wavelet [55] or Fourier [9] transforms. These transforms use predefined basis functions with strong compression property to provide compact (low-rank) description of natural images. The compression property of a basis directly corresponds to the decay rate of the transformed coefficients. The main steps in transform-domain low-rank representation include (i) choosing an appropriate transformation basis (expansion functions), (ii) performing

the forward transformation to obtain the transformed representation of the original parameters, (iii) identifying and retaining only significant coefficients of the transformed representation, and (iv) back transformation to the original domain using only the retained coefficients. The compressive nature of the transforms implies that the transformed representation is sparse, that is, very few of the transformed coefficients are significant. In this section, we present some of the important compressive transforms that have been used for parameterization. The discussion on identifying and retaining the significant elements in the transformed representation is presented in Section 4.

The choice of an appropriate basis to compactly represent model parameters is intimately related to the prior knowledge about the characteristics of the underlying properties of the model, e.g. existing correlation/connectivity structures or possible discontinuous features. In fact, when specific prior models are available, one could construct a specialized transformation that is learned from those models and training data. Examples of specialized transform basis functions that are learned from prior information include the principal component analysis (PCA) [41] and the k-SVD [2] for sparse dictionary learning, which are discussed in this section. In many situations, however, explicit prior models or training data may not be available. In those cases, generic transforms that are used in image compression provide an attractive option for parameterization. We briefly discuss two popular generic transformation methods, namely Fourier transform [9] and its practical and efficient variation known as the discrete cosine transform (DCT) [3, 39] and the wavelet transform [55, 77].

3.2.1 Generic Compressive Transforms

Generic compressive transforms consist of n linearly independent basis vectors in $\mathbb{R}^n$ that can be used to span any length-n vector (or vectorized image). While a complete representation of a length-n parameter vector is possible in a compressive basis, the objective is to obtain an approximate representation by only retaining $k << n$ significant basis elements. Suppose that the set $\{\boldsymbol{\phi}_{i:i=1,\ldots,n}\}$ contains all the basis vectors that are needed for perfect representation in $\mathbb{R}^n$, and a subset $\boldsymbol{\Phi} = \{\boldsymbol{\phi}_{i:i=1,\ldots,k}\}$, with no particular order, provides an acceptable approximation for a vector of interest $\mathbf{u}$. Selection of the subset with k elements depends on the original vector to be approximated and the basis used.

Fourier and Wavelet Transforms Fourier basis functions describe a signal in terms of its frequency content. In this case, if an $n = \prod n_i$ dimensional signal $\mathbf{u}$ is defined in $\mathbb{R}^{n_1 \times n_2 \times \ldots \times n_d}$, the FT at frequency $(f_1, \ldots, f_{n_d})$ will be calculated as:

$$\mathbf{v}(f_1, \ldots, f_{n_d}) = \sum_{i_1=0}^{n_1-1} \cdots \sum_{i_{n_d}=0}^{n_d-1} \mathbf{u}(i_1, \ldots, i_{n_d}) e^{-i2\pi(\sum_{t=1}^{n_d} \frac{f_t i_t}{n_t})} \tag{14}$$

The back transformation that returns **u** can be expressed as:

$$\mathbf{u}(i_1, \ldots, i_{n_d}) = \frac{1}{n} \sum_{f_1=0}^{n_1-1} \cdots \sum_{f_{n_d}=0}^{n_d-1} \mathbf{v}(f_1, \ldots, f_{n_d}) e^{i2\pi(\sum_{t=1}^{n_d} \frac{f_t i_t}{n_t})} \tag{15}$$

If the main features in **u** are captured by low-frequency elements, which is especially true for smooth and correlated vectors, one could approximate **u** by truncating the basis elements with frequencies exceeding a certain threshold. The $(n\text{-}k)$ coefficients corresponding to frequencies higher than the specified threshold are then set to zero.

The DCT is a special case of the Fourier transform that only considers the real part of $e^{-i2\pi(\sum_{t=1}^{n_d} \frac{f_t i_t}{n_t})}$, which is $\cos\{2\pi(\sum_{t=1}^{n_d} \frac{f_t i_t}{n_t})\}$. Hence, the transformation takes the form:

$$\mathbf{v}(f_1, \ldots, f_{n_d}) = \sum_{i_1=0}^{n_1-1} \cdots \sum_{i_{n_d}=0}^{n_d-1} \mathbf{u}(i_1, \ldots, i_{n_d}) \cos\{2\pi(\sum_{t=1}^{n_d} \frac{f_t i_t}{n_t})\} \tag{16}$$

Similar to Fourier transform, an approximation of the original signal **u** is obtained by truncating the frequencies above a certain threshold. Fourier-based transforms can only represent information either in space or frequency domains. This means that once a signal is transformed to Fourier domain, it loses the spatial information and vice versa. Hence, the Fourier basis elements are global and do not encode local information.

Unlike the Fourier transform, the basis elements in Wavelet transform contain both space and frequency information. This implies that each basis vector is localized in space and represents a certain frequency content. Therefore, for any spatial location, one can retain (truncate) specific frequency components that are significant (insignificant). Figure 2(a) and (b) shows 64 sample basis elements for the DCT and Haar wavelet transforms in $\mathbb{R}^{64\times64}$, respectively. As can be verified, the basis images for the DCT transform are not localized in space while those for the discrete Haar wavelet clearly exhibit localized patterns. While generic compressive transforms have useful properties that make them very desirable when explicit prior knowledge is not available, in applications where prior knowledge about the solution (e.g. a training dataset) is available, one may be able to construct more specialized transforms with better performance.

3.2.2 Learned Compressive Transforms

Pre-constructed compressive bases achieve good compression performance in representing smooth and piecewise smooth images when specific knowledge about the image to be compressed is not available. For most natural images only a small subset of the transformed coefficients is sufficient to capture the main features of an image. This implies that most natural images have sparse approximations

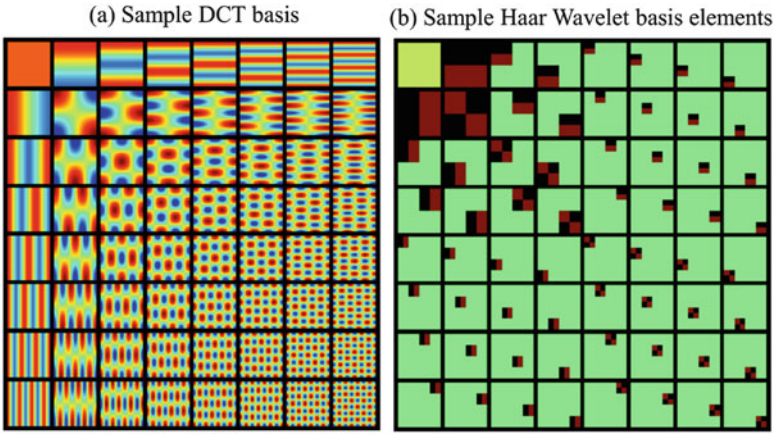

Fig. 2 Examples of generic (pre-computed) compressive transform bases: (a) sample low-frequency basis elements from the DCT basis; (b) sample basis elements from the discrete Haar wavelet. Example is shown for a 64×64 two-dimensional image. The basis elements are separated with black boxes

in these compressive transform domains. However, compressed representation of complex image features with generic transforms may require too many coefficients, which is not desirable for parameterization. Hence, a more sophisticated approach is needed to capture complex features in certain applications. In general, when a specific type of image (e.g. human face) is to be compressed, transforms that are specialized to represent the underlying features are more efficient. For example, in subsurface modelling, where extensive efforts go into data collection and site surveys to construct prior models, specialized transform-domain representations that are tailored to the information in the prior knowledge are more suitable.

Principal Component Analysis (PCA) The PCA [41] is widely used for dimensionality reduction in a wide range of applications. The PCA basis functions capture the main variability and structures in multivariate datasets, which can be exploited in compactly representing/approximating them with minimum loss of information. When the PCA is applied to the covariance matrix of a stochastic process, it diagonalizes the covariance and can be used to define a new (often more desirable) uncorrelated random process. In this case, the PCA provides an orthogonal transformation matrix with decorrelating power that contains, in its columns, the eigenvectors of the covariance matrix. The strong decorrelating property of the PCA basis is advantageous in eliminating parameter correlations (redundancies) to reduce dimensionality. In fact, the PCA sets the standard for dimension reduction with linear transforms as it gives the minimum error (in least-squares sense) in approximating an n-dimensional signal with $S << n$ basis elements (for a fixed S).

The parameterization with PCA follows the same format as in Equation (8), i.e. $\mathbf{u} = \mathbf{\Phi v} = \sum_{i=1}^{k} \boldsymbol{\phi}_i v_i$, where the basis functions ϕ_i's are the eigenvectors of the

covariance matrix of $\mathbf{u}$. Denoting an $n \times 1$-dimensional random variable as $\mathbf{u}$ and its covariance matrix as $\mathbf{C_u}$, the eigenvalue decomposition of the covariance matrix provides the following diagonalization form:

$$\mathbf{C_u} = \mathbf{\Phi \Lambda \Phi}^T \tag{17}$$

where $\mathbf{\Lambda}$ is a diagonal matrix (with eigenvalues of $\mathbf{C_u}$ in its diagonal entries) and $\mathbf{\Phi}$ is an orthonormal (transformation) matrix that has the eigenvectors of $\mathbf{C_u}$ in its columns. If sample realizations of $\mathbf{u}$ are collected into a data matrix $\mathbf{U}_{n\times L} = [\mathbf{u}_1 \dots \mathbf{u}_i \dots \mathbf{u}_L]$, the sample covariance matrix $\mathbf{C_u}$ can be computed as:

$$\mathbf{C_u} = \frac{1}{L-1}(\mathbf{U} - \bar{\mathbf{u}}\mathbf{1}_{1\times L})(\mathbf{U} - \bar{\mathbf{u}}\mathbf{1}_{1\times L})^T \tag{18}$$

where $\bar{\mathbf{u}}$ denotes the mean of $\mathbf{U}$, that is $\bar{\mathbf{u}} = \frac{1}{L}\sum_{i=1}^{L} \mathbf{u}_i$. The term $\frac{1}{\sqrt{L-1}}(\mathbf{U} - \bar{\mathbf{u}}\mathbf{1}_{1\times L})$ can be expressed in terms of its singular value decomposition (SVD) as [48, 72]:

$$\frac{1}{\sqrt{L-1}}(\mathbf{U} - \bar{\mathbf{u}}\mathbf{1}_{1\times L}) = \mathbf{\Psi \Sigma V}^T \tag{19}$$

where $\mathbf{\Psi}$ and $\mathbf{V}$ are orthonormal matrices that contain the left- and right-singular vectors of $\frac{1}{\sqrt{L-1}}(\mathbf{U} - \bar{\mathbf{u}}\mathbf{1}_{1\times L})$, respectively. Combining (18) and (19) yields

$$\mathbf{C_u} = (\mathbf{\Psi \Sigma V}^T)(\mathbf{\Psi \Sigma V}^T)^T = \mathbf{\Psi \Sigma V}^T\mathbf{V \Sigma \Psi}^T = \mathbf{\Psi \Sigma}^2\mathbf{\Psi}^T \tag{20}$$

which reveals $\mathbf{\Psi} = \mathbf{\Phi}$; that is, the left-singular vectors of $\frac{1}{\sqrt{L-1}}(\mathbf{U} - \bar{\mathbf{u}}\mathbf{1}_{1\times L})$ are identical to the eigenvectors of the sample covariance matrix $\mathbf{C_u}$. This relation shows that for high-dimensional variables the PCA transformation matrix can be more efficiently computed by obtaining the left-singular vectors of $\frac{1}{\sqrt{L-1}}(\mathbf{U} - \bar{\mathbf{u}}\mathbf{1}_{1\times L})$, which is computationally more efficient for large n. One could therefore see the correspondence between the left-singular vectors of the sample data matrix and the eigenvectors of the data covariance [30]. It can be shown that amongst all S-term (rank-S) linear approximations of $\mathbf{U}$ the expansion using its S leading left-singular vectors (denoted as $\mathbf{\Phi}_{n\times S}$) gives the smallest root-mean-square error (RMSE).

Sparse Dictionary Learning (k-SVD) While PCA offers a very efficient decorrelating basis for compact representations, it is a linear transform in which the significant basis elements are predetermined and fixed. Recent developments in sparse signal processing have led to growing interest in sparse dictionary learning algorithms. A major distinction between PCA and sparse dictionaries is in the way the significant elements are selected. In sparse reconstruction, the significant elements are neither predetermined (ranked) nor fixed; rather, they must be identified independently for each instance of the parameter vector.

For construction of sparse dictionaries from a training dataset with L elements, $\mathbf{U}_{n\times L} = [\mathbf{u}_1 \dots \mathbf{u}_i \dots \mathbf{u}_L]$, one can solve either of the following optimization problems [2, 44, 82]:

$$\min_{[\mathbf{v}_1\mathbf{v}_2\dots\mathbf{v}_L],\boldsymbol{\Phi}} \quad \|\mathbf{v}_i\|_0 \qquad \text{s.t.,} \qquad \sum_{i=1}^{L} \|\mathbf{u}_i - \boldsymbol{\Phi}\mathbf{v}_i\|_2^2 \le \epsilon \quad \text{for} \quad i \in 1:L \tag{21a}$$

$$\min_{[\mathbf{v}_1\mathbf{v}_2\dots\mathbf{v}_L],\boldsymbol{\Phi}} \quad \sum_{i=1}^{L} \|\mathbf{u}_i - \boldsymbol{\Phi}\mathbf{v}_i\|_2^2 \qquad \text{s.t.,} \qquad \|\mathbf{v}_i\|_0 \le S \quad \text{for} \quad i \in 1:L \tag{21b}$$

where $\|\mathbf{v}_i\|_0$ refers to the number of non-zero entries in $\mathbf{v}_i$ (i.e. S). Equations (21a) and (21b) are alternative formulations for sparse dictionary learning. In Equation (21a), a maximum allowable representation error is used as a constraint while the level of sparsity for each realization of the prior model is minimized. In Equation (21b), the level of sparsity is constrained while minimizing the approximation error to represent each realization. Finding the exact solution to the problems in (21) is intractable. However, heuristic methods, such as the k-SVD algorithm, provide practical approximate solutions. We note that in our notation S refers to the sparsity level (number of significant elements retained in the approximation), and k is the dictionary size (total number of dictionary elements), with $S << k$. We briefly describe the k-SVD algorithm as one approach to learn sparse geologic dictionaries from a set of prior training models (more details can be found in the original publications [2]). The k-SVD algorithm takes its name from the k-means clustering algorithm. While the latter computes k mean values at each iteration, the former applies k SVD operations at each iteration. The k-SVD algorithm constructs a dictionary $\boldsymbol{\Phi}$ with size $n \times k$ from L samples of $\mathbf{u}_i$, while ensuring that the projection of each $\mathbf{u}_i$ on $\boldsymbol{\Phi}$ is S-sparse, a problem that is formalized in Equation (21). We also note that for model reduction and approximation purposes, we consider under-complete dictionaries, where exact representation may not be achievable. However, the resulting representation can provide close approximations in a very low-dimensional space.

To construct $\boldsymbol{\Phi}$ and $\mathbf{V}$ from $\mathbf{U}$, the k-SVD algorithm iteratively solves the problem specified in (21). Each iteration of the algorithm consists of two steps: Step 1, sparse coding, used to find the sparse representations (i.e. $\mathbf{V}$) for the entire prior library by fixing $\boldsymbol{\Phi}$; and Step 2, dictionary updating, which finds a new $\boldsymbol{\Phi}$ after fixing the sparse representation $\mathbf{V}$ from Step 1. These two basic steps in the k-SVD algorithm are summarized in Appendix 1. While no formal convergence proof has been given for this algorithm, numerical experiments show that it is generally robust [2, 44, 45]. It is important to note that the k-SVD algorithm is computationally demanding, especially when the dimension of the dictionary increases. Each iteration of the k-SVD algorithm requires k orthogonal matching pursuit (OMP) [83] sparse coding and k rank-one SVD operations, both are computationally expensive operations. However, the computations related to construction of a sparse dictionary are

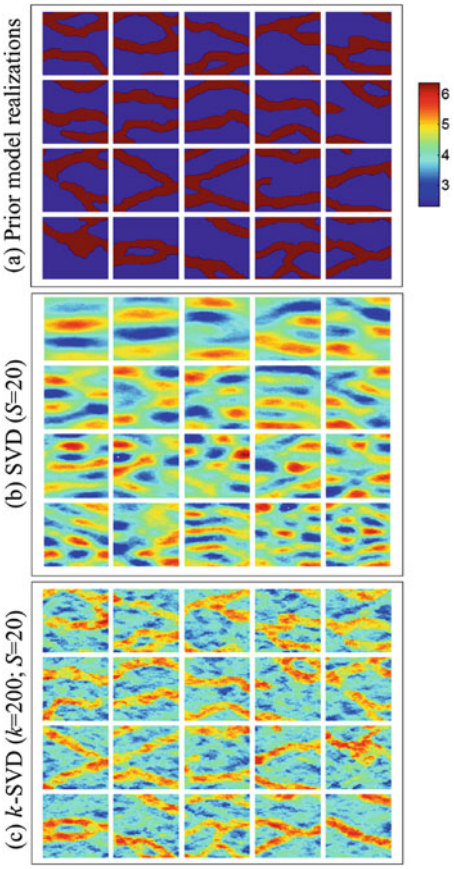

Fig. 3 Examples of learned expansion images using prior training data: (a) prior (training) models used for constructing linear expansion images; (b) $S = 20$ leading PCA basis elements; and (c) sample k-SVD dictionary elements with $S = 20$ and $k = 200$. Examples are shown for $n_x \times n_y = 100 \times 100$ two-dimensional model. The images are separated using white borders

performed offline and can be considered as part of the training step. In addition, the original k-SVD algorithm is typically applied to obtain over-complete dictionaries for small image segments [2]. For large-scale inverse problems, the method has been used to obtain under-complete dictionaries [44, 45].

Figure 3 shows an example of dictionary learning in geosciences applications. Figure 3(a) depicts samples from the training data that represent two-dimensional fluvial channel configurations (generated using SNESIM conditional simulation algorithm [75]). In this figure, the red regions represent fluvial channels that are composed of sandstone with very high rock permeability (to fluid flow) values while the blue regions describe shale or mudstone with very low-permeability values. The high-permeability values manifest their importance in fluid flow and displacement patterns by creating preferential flow patterns within the channel regions. Figure 3(b) presents the first $S = 20$ PCA basis (Eigen) images that

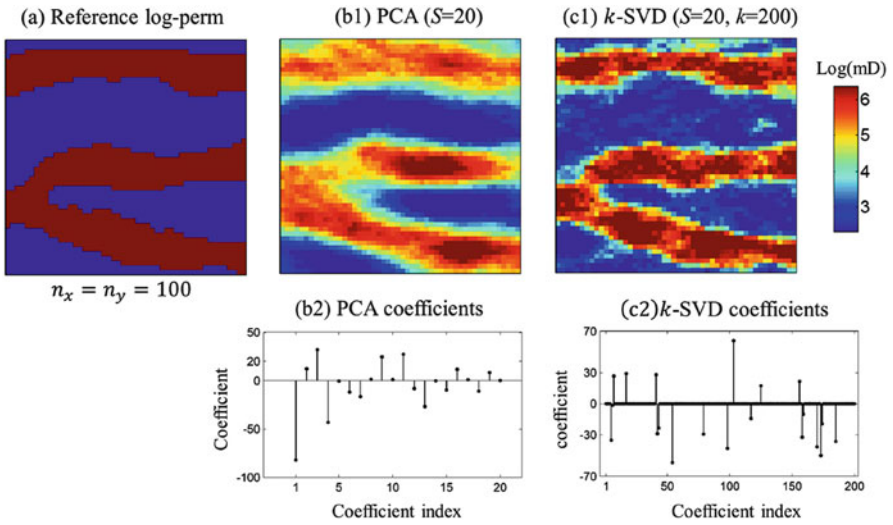

Fig. 4 Compression performance of the PCA and k-SVD: (a) a sample image similar to the training data; (b1)–(b2) results of compressed representation with $S = 20$ leading PCA basis images and the corresponding PCA coefficients, respectively; and (c1)–(c2) results of compressed representation using the k-SVD with $S = 20$ and $k = 200$ and the corresponding k-SVD coefficients

correspond to this training data, and Figure 3(c) shows the corresponding sample elements from the k-SVD dictionary, using $S = 20$ and $k = 200$. To illustrate the approximation performance of the PCA and k-SVD, Figure 4(a) depicts a model that is structurally similar to those in the training data, along with its PCA and k-SVD approximations in Figure 4(b1) and (c1), respectively, using $S = 20$. The corresponding transform coefficients for each case are shown in the second row (Figure 4(b2) and (c2)). Figure 4(b2) and (c2) shows a major difference between the PCA and k-SVD representations, which is the sorting of the PCA elements that leads to identification of $S = 20$ fixed elements. In case of k-SVD, the significant elements are not predetermined. Instead, the significant elements and their corresponding coefficients are identified by searching through a larger set ($k = 200$) of dictionary elements. However, the selection of significant dictionary elements is not trivial and is usually accomplished through a sparsity-promoting optimization algorithm, generally known as sparse reconstruction.

4 Sparse Reconstruction

Selecting a small subset of dictionary elements out of a large set is posed as a sparse reconstruction problem. A signal $\mathbf{v} \in \mathbb{R}^k$ is considered sparse if a large fraction of its entries are (approximately) zero [5]. A signal is S-sparse if it has at most S non-

zero entries. A signal that may not appear as sparse (in space or time) may have a sparse representation in a different (transform) domain. For instance, in many cases a parameter vector $\mathbf{u}$ may not be sparse but can have a sparse representation $\mathbf{v}$ after transformation through $\mathbf{\Phi}$, that is $\mathbf{u} = \mathbf{\Phi v}$.

Depending on the application, identification of significant dictionary elements can be based either on complete (e.g. image compression [80]) or incomplete knowledge (inverse problem) about the unknown parameters. In inverse problems, often limited measurements are available for identification of the significant dictionary elements, and estimation of their corresponding expansion coefficients. Compressed sensing (also called compressive sensing or compressive sampling) [4, 11, 21] is a relatively new paradigm that provides an alternative to the well-known Shannon sampling theory. Compressed sensing adopts sparsity as prior knowledge about signals, while Shannon theory was designed for frequency band-limited signals. The widespread application of compressed sensing is, in part, due to the universality of the sparsity property that is encountered in a wide range of natural phenomena (especially images). In many cases, sparsity may not be immediately apparent and certain manipulation (e.g. transformations) of the original parameters may be necessary for their sparsity to emerge. For instance, natural images that have various elements with spatial correlations in them do not exhibit sparsity in the space domain but are highly compressible and are well-known to have sparse representation in the Wavelet or DCT domains. One of the main contributors to the widespread application of compressed sensing is its direct application to solving underdetermined inverse problems, such as tomographic image reconstruction [15].

Compressed sensing gives a strong theoretical support and an efficient solution algorithm (under appropriate conditions) for solving otherwise intractable (NP-hard) inverse problems that have sufficiently sparse solutions. To recover a sparse solution $\mathbf{v}$ from a set of linear measurements $\mathbf{d} = \mathbf{Gv}$, one can solve the following minimization problem:

$$\min_{\mathbf{v}} \quad \|\mathbf{v}\|_0 \qquad \text{s.t.,} \qquad \mathbf{d} = \mathbf{Gv} \tag{22}$$

where $\|\mathbf{v}\|_0$ is the ℓ_0-norm (note that ℓ_0 does not conform to norm definition and is often loosely referred to as a norm) of vector $\mathbf{v}$ and represents its cardinality. In this formulation, the optimization problem searches for a solution that reproduces the observed data (constraint) while having a minimum number of non-zero entries (support). The ℓ_0-norm is not a differentiable function and does not lend itself to solution with standard gradient-based optimization methods. In practice, two types of approximate algorithms have been developed to solve (22): (i) greedy pursuit algorithms, such as OMP [83], COSAMP [61], IHT [7], or IMAT [56], and (ii) convex approximations, in which the non-convex ℓ_0-norm is replaced with its convex relaxations, e.g. ℓ_1-norm in basis pursuit [17] or a heuristically defined exponential norm in [58].

Compressed sensing derives the solution by replacing the ℓ_0-norm with ℓ_1-norm and offers conditions under which an exact solution to the original problem is

guaranteed (see [10, 21] for details). In this case, the optimization problem takes the form:

$$\min_{\mathbf{v}} \quad \|\mathbf{v}\|_1 \qquad \text{s.t.,} \qquad \mathbf{d} = \mathbf{G}\mathbf{v} \tag{23}$$

The fundamental importance of this formulation is that it converts the problem from an NP-hard problem to a linear programming problem, which can be solved efficiently. In practice, it can be demonstrated that the ℓ_p-norm, for $0 \leq p \leq 1$, while non-convex, has a similar sparsity-promoting property; however, in addition to solution complexity, the mathematical proof and the required conditions for this case are not well understood.

In many applications, the conditions required to guarantee exact solution may not be met. A particular example of departure from those conditions, which is often encountered in physical systems, is when the measurements are not adequate or the measurement operator is nonlinear. In those cases, it may still be possible to exploit the sparsity-promoting property of the ℓ_1-norm to formulate and solve an inverse problem. The selection property of the ℓ_1-norm penalty offers an important regularization form that can be used to enhance the solution of nonlinear inverse problems when applicable. When the measurement equations are nonlinear, the resulting sparse reconstruction problem takes the form:

$$\min_{\mathbf{v}} \quad \|\mathbf{v}\|_1 \qquad \text{s.t.,} \qquad \|\mathbf{d} - \mathbf{g}(\mathbf{v})\|_2^2 \leq \sigma^2 \tag{24}$$

where $\mathbf{g}(\mathbf{v})$ is a nonlinear operator. Appendix 2 discusses an iteratively reweighted least-squares (IRLS) algorithm for solving the ℓ_1-norm regularized minimization problem. In the next section, we discuss the application of sparsity regularization under nonlinear measurements in subsurface flow and transport inverse problems. In addition to ℓ_1-norm regularization, we will also present the use of a mixed ℓ_1/ℓ_2-norm [23, 40], which is known as group sparsity. When the signal of interest $\mathbf{v}$ has block-sparse behaviour, the ℓ_1/ℓ_2-norm can have a superior reconstruction performance compared to the standard ℓ_1-norm. In block-sparse signals, the entries are collected in predefined groups and the sparsity penalty is applied across the groups. In this case, the ℓ_2-norm is applied to the elements inside each group to quantify the group contribution, and the ℓ_1-norm operates on the computed ℓ_2-norm of the groups to impart sparsity. Mathematically, if $\mathbf{v}_i$'s are subsets of $\mathbf{v}$ and $\bigcup_i \mathbf{v}_i = \mathbf{v}$, then the ℓ_1/ℓ_2-norm is defined as $\|\mathbf{v}\|_{1,2} = \sum_i \|\mathbf{v}_i\|_2$. In this case, the inverse problem formulation minimizes the ℓ_1/ℓ_2-norm of the solution while honouring the measurement constraint, that is:

$$\min_{\mathbf{v}} \quad \|\mathbf{v}\|_{1,2} \qquad \text{s.t.,} \qquad \mathbf{d} = \mathbf{G}\mathbf{v} \qquad \text{linear} \tag{25a}$$

$$\min_{\mathbf{v}} \quad \|\mathbf{v}\|_{1,2} \qquad \text{s.t.,} \qquad \mathbf{d} = \mathbf{g}(\mathbf{v}) \qquad \text{nonlinear} \tag{25b}$$

An example application of group sparsity is presented in the next section. In this case, the objective is to select a small set of the groups within $\mathbf{v}$ that have significant contribution to the solution. In other words, the sparsity is applied to the groups and not individual entries.

5 Subsurface Flow Inverse Modelling

Fluid flow and transport in underground porous rock formations plays a key role in developing the related energy and water resources in these systems. Mathematical modelling of the underlying physical processes is commonly used to predict the response of these systems to perturbations (forcing) introduced during resource development (extraction or injection of fluids). The description of the physical processes that take place in these systems leads to high-dimensional and coupled nonlinear PDEs, which include various rock properties as spatially distributed unknown parameters. It is common to formulate inverse problems to estimate the unknown parameters of these PDEs from observations of the dynamical response of these systems. In this section, we describe the formulation of the related PDE-constrained inverse problem and provide examples to demonstrate their practical application.

5.1 Subsurface Flow Forward Modelling

An important example of PDE-constrained inverse problems is the multi-phase flow equations in the subsurface environments. The spatiotemporal evolution of multi-phase fluid flow can be expressed as a special form of the Navier-Stokes equations [19, 20]. Conservation of mass, momentum, and energy are three fundamental principles in the Navier-Stokes equations, which yield the following PDEs, respectively:

$$\frac{\partial \rho}{\partial t} + \nabla.(\rho \mathbf{v}) = 0 \tag{26a}$$

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v}.\nabla)\mathbf{v} = -\frac{1}{\rho}\nabla P + \mathbf{F} + \frac{\mu}{\rho}\nabla^2 \mathbf{v} \tag{26b}$$

$$\rho\left(\frac{\partial E}{\partial t} + \mathbf{v}.\nabla E\right) - \nabla.(K_H \nabla T) + \rho P \nabla.\mathbf{v} = 0 \tag{26c}$$

where $\mathbf{v}$, E, P, T, ρ, μ, K_H, and $\mathbf{F}$ correspond to velocity, internal thermodynamic energy, pressure, temperature, density, viscosity, heat conduction coefficient, and external forces per unit mass. Here, we consider a special case involving two-phase incompressible and immiscible fluid flow system, for which the governing PDE

Table 1 A summary of physical properties and their definition

Property	Definition
Phase mobility	The ratio of effective permeability to phase viscosity
Phase density	The density of fluids, i.e. oil or water
Formation volume factor	Volume of the phase at the in-situ pressure to its volume at standard surface condition
Permeability	Ability for fluids (gas or liquid) to flow through porous rocks
Porosity	Ratio of void space to total rock volume
Phase saturation	Ratio of pore volume occupied by specific fluid phase
Flux	Flow rate per unit area
Pore volume	Total void volume of reservoir
Wetting phase	The phase with more tendency to maintain contact with the solid surface

equations are expressed by combining mass balance and Darcy's law (representing the momentum balance) [16, 22] as:

$$\nabla.(\frac{\lambda_{\mathrm{w}}}{B_{\mathrm{w}}}\mathbf{u}(\nabla P_{\mathrm{w}} - \gamma_{\mathrm{w}}\nabla Z)) = \frac{\partial}{\partial t}(\phi\frac{S_{\mathrm{w}}}{B_{\mathrm{w}}}) + q_{\mathrm{w}} \tag{27a}$$

$$\nabla.(\frac{\lambda_{\mathrm{n}}}{B_{\mathrm{n}}}\mathbf{u}(\nabla P_{\mathrm{n}} - \gamma_{\mathrm{n}}\nabla Z)) = \frac{\partial}{\partial t}(\phi\frac{S_{\mathrm{n}}}{B_{\mathrm{n}}}) + q_{\mathrm{n}} \tag{27b}$$

In the above equations, w and n represent the wetting and non-wetting phases, and λ, γ, B, $\mathbf{u}$, ϕ, Z, S, and q correspond to the phase mobility, phase density, formation volume factor, intrinsic rock permeability, rock porosity, gravity potential, phase saturation, and flux, respectively (see Table 1 for definitions). The governing equations in Equation (27) involve four unknown dynamic state variables: P_{n}, S_{n}, P_{w}, and S_{w}. Two additional equations are needed to close the PDE system. These two equations are the constitutive equations on the pressures and saturations and are typically expressed as:

$$P_{\mathrm{n}} - P_{\mathrm{w}} = P_c(S_{\mathrm{w}}) \tag{28}$$

$$S_{\mathrm{w}} + S_{\mathrm{n}} = 1 \qquad 0 \leq S_{\mathrm{w}}, S_{\mathrm{n}} \leq 1 \tag{29}$$

The first equation describes the capillary pressure (difference between non-wetting and wetting phase pressures) as a function of the wetting phase saturation (see Table 1 for definition) [49], while the second equation imposes a physical constraint on the saturation of two phases in a fully saturated porous medium.

With specified rock and fluid properties, initial and boundary conditions, and other input parameters and control forcing, the coupled PDE system can be discretized and solved numerically. In practice, the resulting discretized system can be high-dimensional ($\sim 10^{6-7}$ unknowns) and computationally demanding to solve. A simple example of immiscible two-phase flow, in which water is injected to displace oil, is depicted in Figure 5. Figure 5(a) shows a two-dimensional

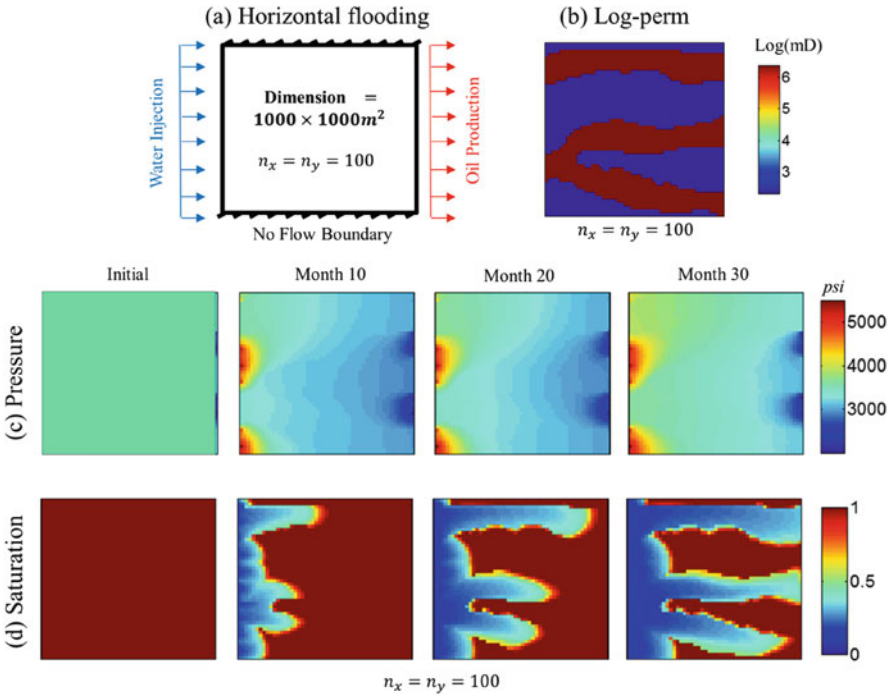

Fig. 5 The forward simulation model used in Example 1: (a) schematic of a reservoir with injection (production) wells on the left (right) side of the domain; (b) the intrinsic permeability distribution in the reference model consisting of high-permeability fluvial channels (red) and low-permeability background shale (blue); and snapshots of pressure (c) and saturation (d) profiles after 10, 20, and 30 months

($1000 \times 1000\,\mathrm{m}^2$) reservoir, which is discretized into 100×100 cells of the same size. A series of water injection wells are placed on the left side of the domain to displace the hydrocarbons toward a similar array of production wells placed on the right side. In this example, the capillary pressure is set to zero everywhere, that is $P_{\mathrm{n}}(x,t) = P_{\mathrm{w}}(x,t)$. Figure 5(b) depicts the intrinsic permeability distribution for this model, which shows a fluvial channel system with high-permeability (red) channels embedded in low-permeability (blue) background shale. As shown in the saturation plots of Figure 5(c), fluids move faster in the high-permeability channel sections. Figure 5(c) and (d) displays the solution of the PDE system as snapshots of pressure and saturation (S_{n}) fields at different times within the first 30 months of the simulation. In our example, the configuration includes the production wells (on the right) that produce water and oil, and injection wells (on the left) that inject water into the reservoir. Initially, the reservoir is fully saturated with the non-wetting phase (oil). Water injection into the reservoir displaces the oil from the left side toward the production wells on the right side, where the mixture of oil and water is extracted.

The forward simulation described above is used to predict the spatiotemporal evolution of the dynamical states (pressure and saturation distributions) of the

system for a given set of input parameters and controls. The state variables of the system are only observable through indirect measurements (e.g. flowrates and pressures) at scattered well locations. The related inverse problem can then be posed to find the unknown parameters (e.g. rock flow properties) from their limited, indirect, and nonlinear measurements.

5.2 Subsurface Flow Inverse Problem

Calibration of subsurface flow forward models against nonlinear dynamic data, i.e. data that are measured at different times and are nonlinearly related to parameters of interest, is commonly used to update model parameters and improve future model predictions [36]. Examples of dynamic data include time series of pressure or fluid flowrate measurements made at the well locations and differential images of fluid saturations typically obtained from seismic surveillance. In particular, dynamic data carry important information about heterogeneous rock flow properties, such as permeability distribution. The difficulty and cost associated with direct sampling from deep geologic formations necessitate the use of subjective assumptions and interpolations, which introduce significant uncertainty in the constructed rock properties. Calibration against dynamic data is a routine task performed to improve the description of these models (e.g. [44, 46]) and the related future forecasts. Dynamic flow data from scattered wells often contain spatially averaged information and offer limited resolution. Therefore, using high-resolution detailed models for unknown parameters can lead to discrepancy between data and model resolutions, a major contributor to the problem ill-posedness.

Prior models of parameters play a significant role in subsurface flow inverse modelling and are commonly used to constrain the inverse modelling solution. Of particular prominence in describing rock flow properties is the type and connectivity of geologic patterns that are expected in a given formation [75, 88]. Even qualitative knowledge about the depositional environment and the type of geologic features can be useful in eliminating implausible solutions. However, in solving the related inverse problems, it is important to acknowledge and reflect the uncertainty in the conceptual models of geologic continuity [25, 28, 43]. In this section, we present subsurface flow inverse modelling formulations that are developed by exploiting the selection property of the sparsity-promoting formulations that were discussed above.

We first consider the same setup in the forward model of Figure 5 and use the PCA and k-SVD representations to solve the corresponding inverse problem. A total of 2000 prior model realizations are generated using geostatistical simulation (Figure 3(a) shows 20 samples). The corresponding PCA and k-SVD basis images are shown in Figure 3(b) and (c), respectively. The ℓ_1-norm regularized formulation is applied to the k-SVD representation while a traditional parameterization using 20 leading basis elements is used for the PCA solution. More specifically, in the case of k-SVD the regularization term $J(\mathbf{v}) = \|\mathbf{v}\|_1$ is minimized along with the data

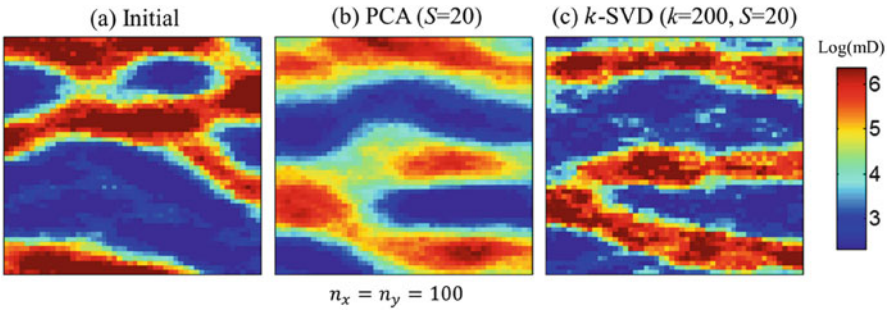

Fig. 6 Solution of the inverse problem in Example 1: (a) initial log-permeability distribution before data integration; (b) estimated log-permeability distribution using the PCA parameterization with $S = 20$; and (c) reconstructed log-permeability distribution using k-SVD with $S = 20$ and $k = 200$. The reference model is shown in Figure 5(b)

mismatch norm $\|\mathbf{d} - \mathbf{g}(\mathbf{u})\|_2^2$. In the case of PCA, the leading $S = 20$ basis elements are selected a-priori and used as parameterization basis vectors; hence, during inversion the coefficients corresponding to these elements are estimated without using the ℓ_1-norm regularization term. The initial model for the inversion is shown in Figure 6(a). The reconstruction results using the PCA and k-SVD descriptions are shown in Figure 6(b) and (c), respectively. The results show better estimation quality with the k-SVD representation and sparse reconstruction algorithm. The improved performance can be attributed to several factors, including flexibility in identifying the low-rank subspace during inversion (PCA provides a predetermined subspace), and better representation of geologic patterns that are not amenable to covariance-based description used in the PCA parameterization. Figure 7 depicts the data match and predictions obtained from the two methods, which seem to be comparable. It is important to note that while the two methods produce similar data matches, the solution from the k-SVD algorithm is visibly superior. This can be understood by recognizing the ill-posed nature of the problem, which implies that many solutions can be found to match the observed data. In this case, the k-SVD representation is better able to capture the connectivity structure in the prior model and the sparse reconstruction algorithm can recover the correct structure more accurately.

5.3 Uncertainty in Initial Geologic Scenario

Prior models of geologic continuity describe the type of geologic patterns and their connectivity. When used as prior model, geologic connectivity carries important weight in finding a solution to subsurface flow inverse problems. However, describing the exact form of connectivity from limited available data is a subjective process and depends on the geologist's interpretation. In generating a subsurface flow model, the connectivity patterns are typically constructed by integrating quantitative (well

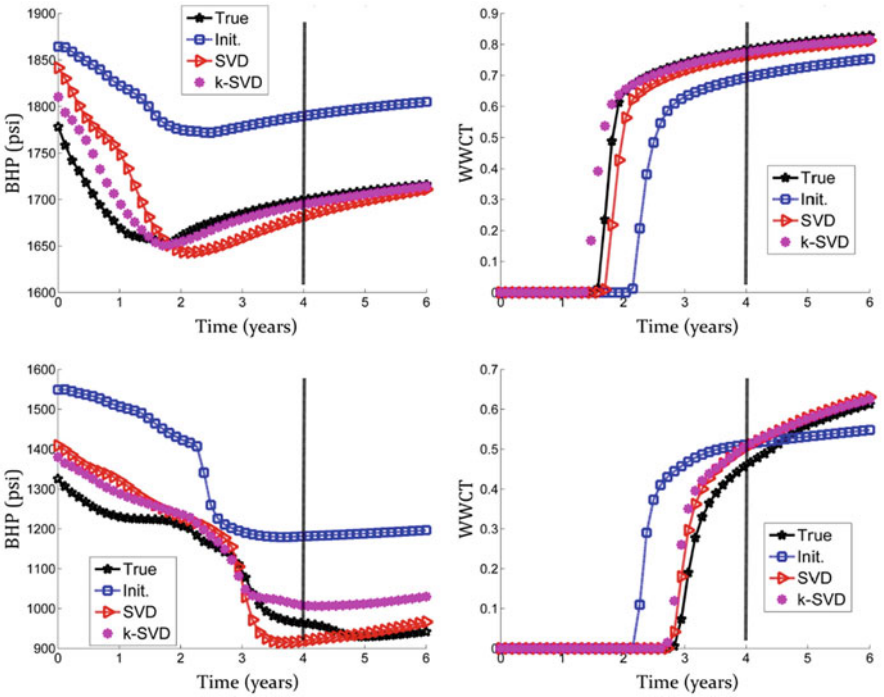

Fig. 7 Sample well data match (first four years) and prediction (last two years) results for Example 1. The BHP and watercut observations at two sample production wells are shown (the producers are under total production rate control)

log, core analysis, and seismic data) and qualitative information (e.g. outcrops) with expert knowledge and interpretation as well as process-based geologic modelling of the depositional environment. Traditionally, a single conceptual model of continuity (e.g. variogram model) is constructed and used to constrain the solution of the inverse problem, assuming perfect knowledge about the continuity model. However, a major source of uncertainty is related to the adopted conceptual model of geologic continuity. Adhering to a single conceptual geologic scenario can lead to underestimation of the initial uncertainty in the prior models and result in solutions that depend heavily on the quality of the adopted prior model (which can be questionable) [25, 28, 38, 43, 46, 67, 68, 73, 76].

Another important implication of adopting a single geologic scenario is eliminating the opportunity to confirm, reject, or correct a proposed geologic scenario based on dynamic data. In generating or selecting the prior geologic scenario, dynamic data is typically not included (usually dynamic data are obtained at later stages). A simple way to address this issue is to include multiple plausible geologic scenarios as possible prior models [25, 28, 38, 43]. These alternative geologic scenarios could be developed as independent interpretation of existing data by different geologists (experts), or they could be derived from a stochastic process-

based geologic modelling framework. Inverse modelling can then be applied to evaluate the plausibility of the proposed geologic scenarios based on available dynamic data. Inversion methods that can incorporate multiple geologic scenarios are not widely studied in the literature. In this section, we present one such inversion method by exploiting the selection property of sparsity-promoting regularization techniques, or more specifically the group-sparsity regularization.

The group-sparsity regularization is implemented by minimizing the ℓ_1/ℓ_2-norm to identify relevant geologic scenarios (from a list of proposed scenarios) based on dynamic flow-related data. Consider p alternative geologic scenarios, each used to generate L different realizations as prior models; that is, $\mathbf{U}_1 = [\mathbf{u}_{11}\mathbf{u}_{12}\dots\mathbf{u}_{1L}], \dots, \mathbf{U}_p = [\mathbf{u}_{p1}\mathbf{u}_{p2}\dots\mathbf{u}_{pL}]$ are p sets of prior model realizations in which the columns of $\mathbf{U}_i = [\mathbf{u}_{i1}\mathbf{u}_{i2}\dots\mathbf{u}_{iL}]$ represent L realizations from the i^{th} geologic scenarios. If the prior model realizations for each scenario are used to generate p different PCA bases, then a hybrid dictionary can be constructed to include all the bases $\mathbf{\Phi} = [\mathbf{\Phi}_1\mathbf{\Phi}_2\dots\mathbf{\Phi}_p]$. Here, the realizations $\mathbf{U}_i$ for each geologic scenario are used to generate a corresponding PCA basis $\mathbf{\Phi}_i = [\boldsymbol{\phi}_{i1}\dots\boldsymbol{\phi}_{is_i}]$, where s_i is the size of low-rank representation. Using this hybrid dictionary, the parameter of interest $\mathbf{u}$ is approximated through a linear expansion of the form $\mathbf{u} = \mathbf{\Phi}\mathbf{v} = [\mathbf{\Phi}_1\mathbf{\Phi}_2\dots\mathbf{\Phi}_p][\mathbf{v}_1;\mathbf{v}_2;\dots;\mathbf{v}_p]$. This formulation implies that all prior geologic scenarios have a chance to represent the solution. However, the underlying assumption is that many of the included prior scenarios are not relevant and should not contribute to reconstruction of the solution. Hence, only very few (if not just one) of the groups are expected to have non-zero weights.

Using a mixed ℓ_1/ℓ_2-norm for group sparsity [29, 54], the regularized objective function of the inverse problem can be expressed as:

$$\begin{aligned} &\min_{\mathbf{v}} \quad J(\mathbf{v}) = \sum_{i=1}^{p} \|\mathbf{v}_i\|_2 \quad \text{s.t.,} \quad \|\mathbf{d} - \mathbf{g}(\mathbf{u})\|_2^2 \le \sigma^2 \\ &\text{and} \qquad \mathbf{u} = \mathbf{\Phi}\mathbf{v} = [\mathbf{\Phi}_1\mathbf{\Phi}_2\dots\mathbf{\Phi}_p][\mathbf{v}_1;\mathbf{v}_2;\dots;\mathbf{v}_p] \end{aligned} \tag{30}$$

After solving this inverse problem, the solution $\mathbf{u}$ and the geologic scenario(s) that significantly contribute to constructing it are identified simultaneously (for more details, see [28]). Appendix 3 presents the details of solving the optimization problem in Equation (30).

The example in this section consists of a numerical two-phase flow in a heterogeneous reservoir for which the intrinsic permeability values in the entire field are unknown. The reservoir model has a dimension of $1000 \times 1000 \times 10\,\mathrm{m}^3$, which is discretized into a $100 \times 100 \times 1$ uniform grid system. Figure 8(a) depicts the configuration of this water-flooding example. An injection well is placed in the middle of the field and eight producers are located along the edges of the reservoir to build a traditional 9-spot water-flooding scheme. A total of 0.8 pore volume (PV) (see Table 1 for definition) of water is injected into the formation during the first 4 years of water flooding. Also, 0.4 PV of water is injected in the following two years,

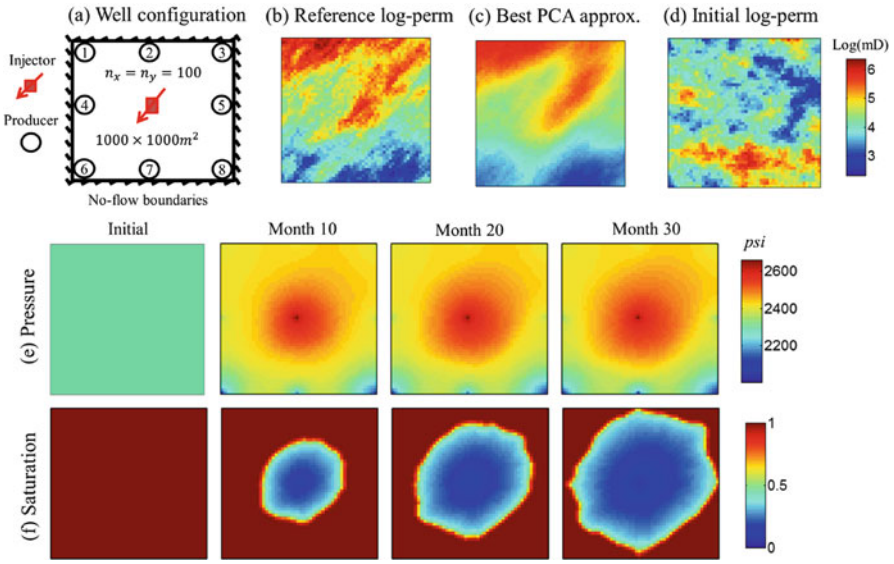

Fig. 8 The forward simulation model used in Example 2: (a) schematic of a reservoir with one injection well in the centre and eight production wells symmetrically distributed along the edges of the domain; (b) the intrinsic log-permeability distribution in the reference model; (c) the best achievable log-permeability estimate when the correct prior geologic scenario is known (Scenario 6); (d) the initial log-permeability distribution, assuming equal contributions from each prior variogram model; and snapshots of pressure (e) and saturation (f) profiles after 10, 20, and 30 months

during the prediction phase. The porosity of the field is assumed to be 0.25 for the entire field, and oil/water viscosity ratio is set to 1. The pressure at the injection well and the total (water and oil) flowrates at the production wells are measured every 40 days and used for inversion. The reference log-permeability map along with its best achievable PCA approximation, and the initial log-permeability map before inversion are shown in Figure 8(b)–(d), respectively. The initial log-permeability map considers equal weight given to all 12 groups. Figure 8(e) and (f) displays the pressure and saturation profiles, respectively, after 10, 20 and 30 months.

To reflect the uncertainty in the prior variogram model, the direction of maximum continuity and the minimum and maximum variogram ranges are assumed to be uncertain. The variogram parameters for these prior models and four samples from the corresponding realizations are shown in Figure 9. Comparing the reference model with the realizations generated using these 12 different prior geologic models reveals that the consistent model belongs to Scenario 6. The projection of the reference map onto the PCA basis corresponding to Scenario 6 is shown in Figure 8(c). Other models either present different directions of global continuity or inaccurate ranges. The realizations from these 12 variogram models are used to build 12 different PCA bases, which are combined to form a hybrid dictionary.

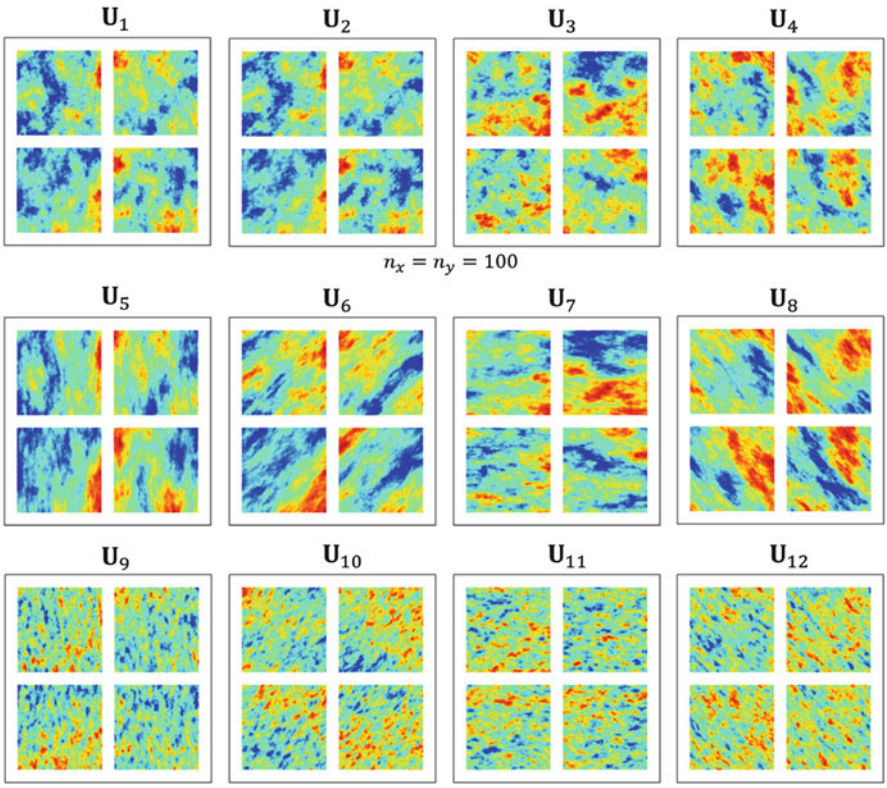

Fig. 9 Alternative prior training data derived from 12 different variogram models; each box contains four sample realization from the training data corresponding to a variogram model; the alternative variogram models are obtained by using three variogram range combinations ($a_{max} = 300m, a_{min} = 240m$), ($a_{max} = 600m, a_{min} = 300m$), and ($a_{max} = 100m, a_{min} = 60m$) and four different azimuth values . The reference model is consistent with Scenario 6 with training data $\mathbf{U}_6$

Figure 10 depicts the inversion solution at different iterations. The initial model (Figure 8(d) and the top row of Figure 10) is projected onto all elements of the hybrid dictionary, and all the prior geologic scenarios equally contribute to the representation of the initial model. The global continuity in the permeability field is captured within the first few iterations. At later iterations, the regularization term fine-tunes the solution by selecting the geologic scenarios that best represent the estimated parameter. Group 6, which has the correct variogram model, has been identified as the most significant prior model (with largest ℓ_2-norm) after convergence of the group-sparsity inversion algorithm. Figure 11 shows a summary of the data match and forecast performance of the solution compared to the initial and reference models in two production wells. The data match and prediction results clearly show the improvements achieved after model calibration.

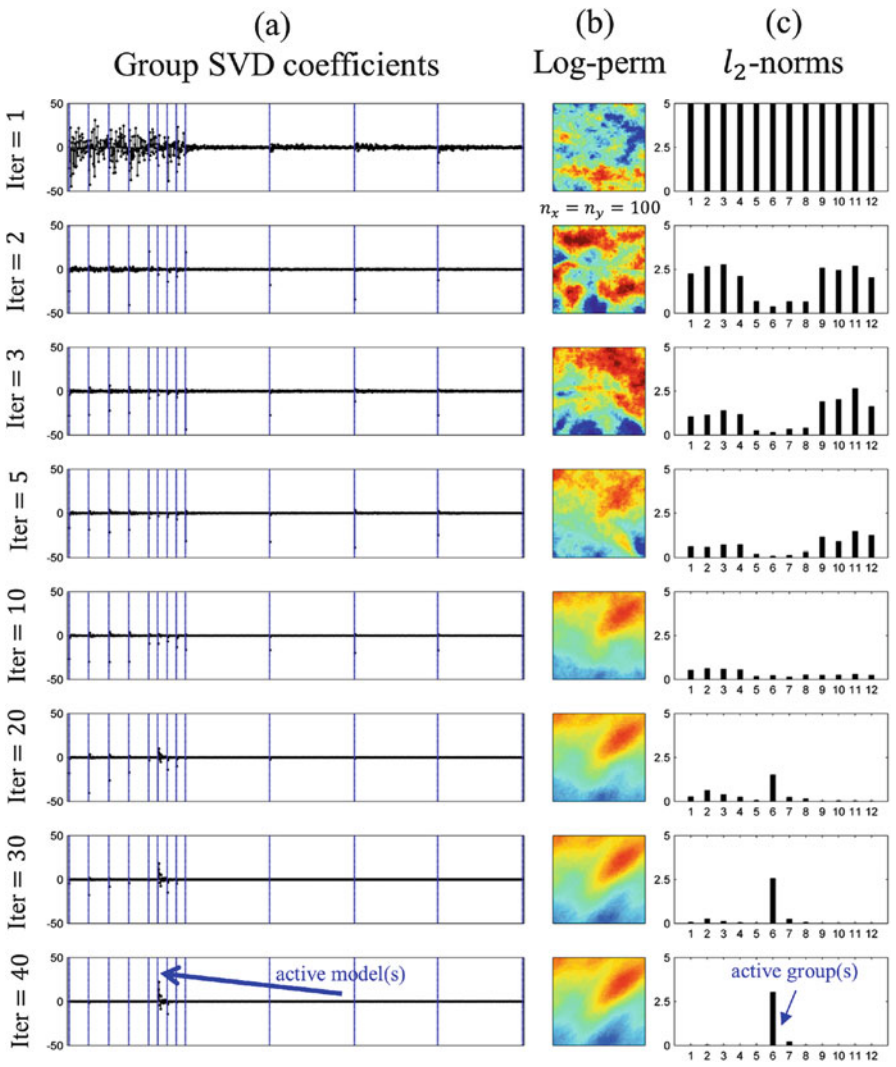

Fig. 10 Results of group-sparsity inversion iterations for Example 2: (a) the coefficients of the expansion using 12 different groups (PCA bases); (b) reconstructed log-permeability map; and (c) the ℓ_2-norm of the coefficients of the PCA representation in each group. Groups with larger ℓ_2-norm have greater contributions to the solution and persist during the iterations, whereas irrelevant groups are assigned insignificant group ℓ_2-norm. Group 6 in this example stay active with a large ℓ_2-norm while other groups disappear during inversion iterations

Two alternative ways may also be used to formulate and solve the above inverse problem: (i) by using the same parameterization, i.e. $\mathbf{\Phi} = [\mathbf{\Phi}_1 \mathbf{\Phi}_2 \dots \mathbf{\Phi}_p]$, with ℓ_1-norm regularization (without group sparsity), and (ii) by combining all the prior models and generating a single PCA parameterization. However, both of

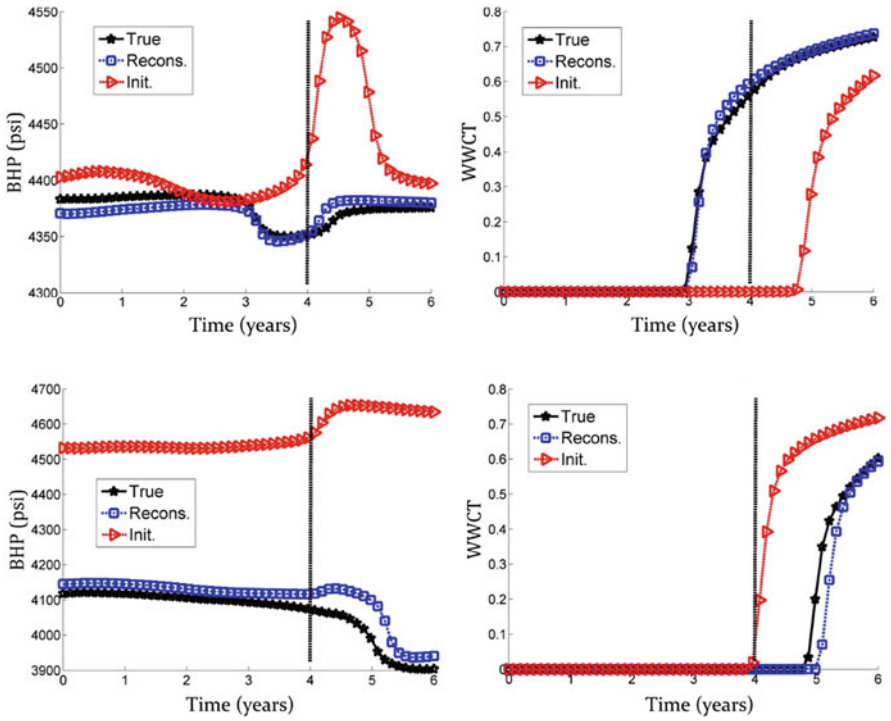

Fig. 11 The pressure and watercut data match (first four years) and forecasts (last two years) for sample production wells. The group-sparsity regularization not only identifies the correct variogram model, it also provides a calibrated model at convergence

these approaches provide inferior solutions. In the first case, the group sparsity, by formulation, has been shown to be more effective in reconstructing the solution as it imposes a stronger constraint on the problem. In the second case, a simple least-square formulation is solved to search for the PCA coefficients in the low-rank subspace defined by the leading PCs, which are not representative of any particular prior (as they represent an aggregate of all prior models).

6 Conclusion

In this chapter, we discussed a general formulation for solving sparse PDE-constrained inverse problems. In particular, we presented sparse inverse problem formulations that use sparsity to regularize ill-posed problems that can arise in various applications. Sparsity is an inherent property of many types of natural images and can be used to improve the solution of ill-posed inverse problems in which the solutions have sparse representations. Examples from multiphase fluid flow in subsurface rock formations, which involve the solution of coupled

PDEs to describe the underling physical processes, were used to demonstrate the effectiveness of the method. To calibrate heterogeneous subsurface flow models against dynamic data, scattered nonlinear measurements of flowrate and pressure are often used. Spatially distributed rock flow properties are known to have a sparse representation in a properly designed basis. High-resolution grid-based description of these properties leads to over-parameterization. When combined with data scarcity, over-parameterized descriptions often lead to problem ill-posedness, introducing great difficulty in solving these inverse problems. Furthermore, prior geologic scenarios that are typically used to regularize these ill-posed inverse problems often involve significant uncertainty that should be taken into account in formulating and solving these problems. We propose the use of learned sparse geologic dictionaries and sparsity-promoting regularization functions as powerful and robust approaches to address these issues. Specifically, we present a formulation in which prior models are used as training data to learn sparse representations of rock flow properties. We show that by promoting sparsity through minimization of regular ℓ_1-norm of the solution in the learned k-SVD dictionary (along with minimization of the predicted and observed data mismatch term) a better-posed inverse problem can be obtained to reconstruct complex geologic patterns. In addition, group-sparsity regularization that minimizes a mixed ℓ_1/ℓ_2-norm was used to discriminate against multiple prior geologic scenarios using flow data. An important implication of the latter is that it allows the use of dynamic flow data in selecting, rejecting, and correcting prior geologic scenarios, a novel concept that can improve traditional subsurface flow model calibration workflows.

Acknowledgements The content of this chapter is based on research partially funded by the US Department of Energy, Foundation CMG, and American Chemical Society.

Appendix 1: k-SVD Dictionary Learning

The k-SVD algorithm is used to construct learned sparse dictionaries from a training dataset. The algorithm is similar to the k-means clustering method and is designed to find a dictionary $\mathbf{\Phi} \in \mathbb{R}^{n\times k}$ containing k elements that sparsely represent each of the training samples in $\mathbf{U}_{n\times L} = [\mathbf{u}_1 \dots \mathbf{u}_i \dots \mathbf{u}_L]$. To achieve this goal, the algorithm attempts to solve the following minimization problem:

$$\hat{\mathbf{V}}, \hat{\mathbf{\Phi}} = \text{argmin}_{\mathbf{V},\mathbf{\Phi}} \quad \sum_{i=1}^{L} \|\mathbf{u}_i - \mathbf{\Phi}\mathbf{v}_i\|_2^2 \qquad \text{s.t.,} \qquad \|\mathbf{v}_i\|_0 \leq S \quad \text{for} \quad i \in 1:L \tag{31}$$

where $\mathbf{V}_{k\times L} = [\mathbf{v}_1 \dots \mathbf{v}_i \dots \mathbf{v}_L]$ are the expansion coefficients corresponding to the training data. Given the NP-hard nature of the problem, the k-SVD algorithm uses a heuristic greedy solution technique by dividing the above optimization problem into

Table 2 k-SVD algorithm

Initialization: Initialize dictionary with $\mathbf{\Phi}^{(0)} \in \mathbb{R}^{n\times k}$. Set $j = 1$.
REPEAT until stopping criteria is met
a. Sparse Coding Step:
-Using a pursuit algorithm (e.g. OMP) compute $\mathbf{V}_{k\times L}^{(j)} = [\mathbf{v}_1\mathbf{v}_2 \dots \mathbf{v}_L]$ as the solution of
$\mathbf{V}^{(j)} = \text{argmin}_{\mathbf{v}_i} \quad \|\mathbf{u}_i - \mathbf{\Phi}^{(j-1)}\mathbf{v}_i\|_2^2 \qquad \text{s.t.,} \qquad \|\mathbf{v}_i\|_0 \le S \quad \text{for} \quad i \in 1:L$
b. Dictionary Update Step:
For each column $c = 1, 2, \dots, k$ in $\mathbf{\Phi}^{(j-1)}$
-Define the group of prior model instances that use this element
$\omega_c = \{i \vert 1 \le i \le L, \mathbf{V}^{(j)}(c, i) \ne 0\}$
-Compute the residual matrix $\mathbf{E}_c = \mathbf{U} - \sum_{i\ne c} \boldsymbol{\phi}_i {\mathbf{v}_c}^T$, where ${\mathbf{v}_c}^T$ is the c^{th} row of $\mathbf{V}^{(j)}$
-Restrict $\mathbf{E}_c$ by choosing columns corresponding to ω_c , i.e. find $\mathbf{E}_c^{\omega}$
-Apply rank-1 SVD decomposition $\mathbf{E}_c^{\omega} = \mathbf{A\Delta B}$
-Update the dictionary element $\boldsymbol{\phi}_c = \mathbf{a}_1$ and the sparse representation $\mathbf{v}_c$ by $\mathbf{v}_c^{\omega} = \mathbf{\Delta} \mathbf{b}_1$
-END

two subproblems: (i) sparse coding and (ii) dictionary update. In the sparse coding step, for the current dictionary, a basis pursuit algorithm is used to find the sparse representation for each member of the training dataset. In the dictionary update step, the sparse representation obtained in the first step is fixed and the dictionary elements are updated to reduce the sparse approximation error. These two steps are repeated until convergence. Table 2 summarizes the k-SVD algorithm. Further details about the k-SVD algorithm may be found in [2]. We note that for high-dimensional training data the k-SVD dictionary learning can be computationally expensive. The computational complexity of each iteration of k-SVD is $O(L(2nk + S^2k + 7Sk + S^3 + 4Sn) + 5nk^2)$, where S is the sparsity level. One strategy to improve the computational efficiency of the algorithm includes using segmentation or approximate low-rank representations of the training data (to reduce n).

Appendix 2: IRLS Algorithm

We use the IRLS algorithm [14] to solve the ℓ_1-norm regularized least-square minimization problem, that is:

$$\min_{\mathbf{v}} \quad J(\mathbf{v}) = \|\mathbf{v}\|_1 + \lambda^2 \|\mathbf{d} - \mathbf{g}(\mathbf{\Phi v})\|_2^2 \tag{32}$$

At iteration n of the IRLS algorithm, the ℓ_1-norm is approximated using a weighted ℓ_2-norm as follows:

$$\min_{\mathbf{v}^{(n)}} \quad J(\mathbf{v}^{(n)}) = \sum_i w_i^{(n)} {v_i^{(n)}}^2 + \lambda^2 \|\mathbf{d} - \mathbf{g}(\mathbf{\Phi v}^{(n)})\|_2^2 \tag{33}$$

where $w_i^{(n)} = \frac{1}{(v_i^{(n-1)^2} + \epsilon^{(n)})^{0.5}}$, (n) stands for the iteration n, and $\epsilon^{(n)}$ is a sequence of small numbers (that converge to zero with increasing n). Using this approximation of the objective function, and a first-order Taylor expansion for $\mathbf{g}(\mathbf{\Phi v}^{(n)})$, the objective function in (33) takes the form:

$$\min_{\mathbf{v}^{(n)}} \quad J(\mathbf{v}^{(n)}) = \sum_i w_i^{(n)} {v_i^{(n)}}^2 + \lambda^2 \|\mathbf{d} - \mathbf{g}(\mathbf{\Phi v}^{(n-1)}) - {\mathbf{G_v}}^{(n)}(\mathbf{v}^{(n)} - \mathbf{v}^{(n-1)})\|_2^2 \tag{34}$$

Here, ${\mathbf{G_v}}^{(n)}$ is the Jacobian matrix of $\mathbf{g}(.)$ with respect to $\mathbf{v}$ at $\mathbf{v} = \mathbf{v}^{(n-1)}$. The updated solution at iteration n can be easily found by taking the derivative of the above convex function w.r.t. $\mathbf{v}^{(n)}$ and setting it to zero.

Appendix 3: Group-Sparsity Inversion

The objective function for group-sparsity regularization can be expressed as:

$$\min_{\mathbf{v}} \quad J(\mathbf{v}) = \sum_{i=1}^{p} \|\mathbf{v}_i\|_2 + \lambda^2 \|\mathbf{d} - \mathbf{g}(\mathbf{\Phi v})\|_2^2 \tag{35}$$

where the notations are discussed in the text. At iteration n, using the Gauss-Newton method and the first-order Taylor series for $\mathbf{g}(\mathbf{\Phi v})$, the linearized version of the above function takes the form:

$$\min_{\mathbf{v}^{(n)}} \quad J(\mathbf{v}^{(n)}) = \sum_{i=1}^{p} (\sum_{j=1}^{s_i} ({v_i^j}^{(n)})^2)^{\frac{1}{2}} + \lambda^2 \|\mathbf{d} - \mathbf{g}(\mathbf{\Phi v}^{(n-1)}) - {\mathbf{G_v}}^{(n)}(\mathbf{v}^{(n)} - \mathbf{v}^{(n-1)})\|_2^2 \tag{36}$$

where ${\mathbf{G_v}}^{(n)}$ is the Jacobian matrix of $\mathbf{g}(\mathbf{v})$, and v_i^j is the jth basis in the ith group. Denoting $\mathbf{\Delta d}^{(n)} = \mathbf{d} - \mathbf{g}(\mathbf{\Phi v}^{(n-1)}) + {\mathbf{G_v}}^{(n)}\mathbf{v}^{(n-1)}$, (36) can be simplified to:

$$\min_{\mathbf{v}^{(n)}} \quad J(\mathbf{v}^{(n)}) = \sum_{i=1}^{p} (\sum_{j=1}^{s_i} ({v_i^j}^{(n)})^2)^{\frac{1}{2}} + \lambda^2 \|\mathbf{\Delta d}^{(n)} - {\mathbf{G_v}}^{(n)}\mathbf{v}^{(n)}\|_2^2 \tag{37}$$

The derivative of the regularization term with respect to ${v_i^j}^{(n)}$ can be approximated as:

$$\frac{{v_i^j}^{(n)}}{(\sum_{k=1}^{s_i} ({v_i^k}^{(n)})^2)^{\frac{1}{2}}} \approx \frac{{v_i^j}^{(n)}}{(\sum_{k=1}^{s_i} ({v_i^k}^{(n-1)})^2 + {\epsilon_i}^{(n)})^{\frac{1}{2}}} \tag{38}$$

where ${\epsilon_i}^{(n)}$ is a small positive number that is used to avoid zero denominators. Note that ${v_i^k}^{(n)}$ in the denominator is approximated as ${v_i^k}^{(n-1)}$. Choosing ϵ such that $0 < {\epsilon_i}^{(n)} < {\epsilon_i}^{(n-1)}$ and $\lim_{n\to\infty} {\epsilon_i}^{(n)} = 0$, it can be shown that this approximation does

not change the solution of the original minimization problem. The iterative solution of (37) can now be derived as:

$$(\mathbf{\Lambda}^{(n)} + \alpha {\mathbf{G_v}^{(n)}}^T \mathbf{G_v}^{(n)})\mathbf{v}^{(n)} = \alpha {\mathbf{G_v}^{(n)}}^T \mathbf{\Delta d}^{(n)} \tag{39}$$

where $\alpha = 2\lambda^2$, and $\mathbf{\Lambda}^{(n)}$ is a diagonal matrix with diagonal entries $\frac{1}{(\sum_{k=1}^{s_i}(v_i^{k(n-1)})^2+\epsilon_i{}^{(n)})^{\frac{1}{2}}}$.

References

1. Aanonsen SI, Nævdal G, Oliver DS, Reynolds AC, Vallès B, et al (2009) The ensemble Kalman filter in reservoir engineering–a review. Spe Journal 14(03):393–412
2. Aharon M, Elad M, Bruckstein A (2006) *rmk*-svd: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on signal processing 54(11): 4311–4322
3. Ahmed N, Natarajan T, Rao KR (1974) Discrete cosine transform. IEEE transactions on Computers 100(1):90–93
4. Baraniuk RG (2007) Compressive sensing [lecture notes]. IEEE signal processing magazine 24(4):118–121
5. Berinde R, Gilbert AC, Indyk P, Karloff H, Strauss MJ (2008) Combining geometry and combinatorics: A unified approach to sparse signal recovery. In: Communication, Control, and Computing, 2008 46th Annual Allerton Conference on, IEEE, pp 798–805
6. Bhark EW, Jafarpour B, Datta-Gupta A (2011) A generalized grid connectivity–based parameterization for subsurface flow model calibration. Water Resources Research 47(6)
7. Blumensath T, Davies ME (2009) Iterative hard thresholding for compressed sensing. Applied and computational harmonic analysis 27(3):265–274
8. Boyd S, Vandenberghe L (2004) Convex optimization. Cambridge university press
9. Bracewell RN, Bracewell RN (1986) The Fourier transform and its applications, vol 31999. McGraw-Hill New York
10. Candes EJ (2008) The restricted isometry property and its implications for compressed sensing. Comptes Rendus Mathematique 346(9–10):589–592
11. Candès EJ, Wakin MB (2008) An introduction to compressive sampling. IEEE signal processing magazine 25(2):21–30
12. Carrera J, Neuman SP (1986) Estimation of aquifer parameters under transient and steady state conditions: 1. maximum likelihood method incorporating prior information. Water Resources Research 22(2):199–210
13. Chandrasekaran V, Recht B, Parrilo PA, Willsky AS (2012) The convex geometry of linear inverse problems. Foundations of Computational mathematics 12(6):805–849
14. Chartrand R, Yin W (2008) Iteratively reweighted algorithms for compressive sensing. In: Acoustics, speech and signal processing, 2008. ICASSP 2008. IEEE international conference on, IEEE, pp 3869–3872
15. Chen GH, Tang J, Leng S (2008) Prior image constrained compressed sensing (PICCS): a method to accurately reconstruct dynamic CT images from highly undersampled projection data sets. Medical physics 35(2):660–663
16. Chen S, Doolen GD (1998) Lattice Boltzmann method for fluid flows. Annual review of fluid mechanics 30(1):329–364
17. Chen SS, Donoho DL, Saunders MA (2001) Atomic decomposition by basis pursuit. SIAM review 43(1):129–159
18. Chen Y, Oliver DS (2012) Multiscale parameterization with adaptive regularization for improved assimilation of nonlocal observation. Water resources research 48(4)

19. Chorin AJ (1968) Numerical solution of the Navier-Stokes equations. Mathematics of computation 22(104):745–762
20. Constantin P, Foias C (1988) Navier-stokes equations. University of Chicago Press
21. Donoho DL (2006) Compressed sensing. IEEE Transactions on information theory 52(4):1289–1306
22. Efendiev Y, Durlofsky L, Lee S (2000) Modeling of subgrid effects in coarse-scale simulations of transport in heterogeneous porous media. Water Resources Research 36(8):2031–2041
23. Eldar YC, Kuppinger P, Bolcskei H (2010) Block-sparse signals: Uncertainty relations and efficient recovery. IEEE Transactions on Signal Processing 58(6):3042–3054
24. Engl HW, Hanke M, Neubauer A (1996) Regularization of inverse problems, vol 375. Springer Science & Business Media
25. Feyen L, Caers J (2006) Quantifying geological uncertainty for flow and transport modeling in multi-modal heterogeneous formations. Advances in Water Resources 29(6):912–929
26. Gavalas G, Shah P, Seinfeld JH, et al (1976) Reservoir history matching by Bayesian estimation. Society of Petroleum Engineers Journal 16(06):337–350
27. Gholami A (2015) Nonlinear multichannel impedance inversion by total-variation regularization. Geophysics 80(5):R217–R224
28. Golmohammadi A, Jafarpour B (2016) Simultaneous geologic scenario identification and flow model calibration with group-sparsity formulations. Advances in Water Resources 92:208–227
29. Golmohammadi A, Khaninezhad MRM, Jafarpour B (2015) Group-sparsity regularization for ill-posed subsurface flow inverse problems. Water Resources Research 51(10):8607–8626
30. Golub G, Kahan W (1965) Calculating the singular values and pseudo-inverse of a matrix. Journal of the Society for Industrial and Applied Mathematics, Series B: Numerical Analysis 2(2):205–224
31. Golub GH, Heath M, Wahba G (1979) Generalized cross-validation as a method for choosing a good ridge parameter. Technometrics 21(2):215–223
32. Gómez-Hernánez JJ, Sahuquillo A, Capilla J (1997) Stochastic simulation of transmissivity fields conditional to both transmissivity and piezometric data-i. theory. Journal of Hydrology 203(1–4):162–174
33. Grimstad AA, Mannseth T, Nævdal G, Urkedal H (2003) Adaptive multiscale permeability estimation. Computational Geosciences 7(1):1–25
34. Hansen PC (1992) Analysis of discrete ill-posed problems by means of the l-curve. SIAM review 34(4):561–580
35. Hansen PC (1998) Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion. SIAM
36. Hill MC, Tiedeman CR (2006) Effective groundwater model calibration: with analysis of data, sensitivities, predictions, and uncertainty. John Wiley & Sons
37. Jacquard P, et al (1965) Permeability distribution from field pressure data. Society of Petroleum Engineers Journal 5(04):281–294
38. Jafarpour B, Tarrahi M (2011) Assessing the performance of the ensemble Kalman filter for subsurface flow data integration under variogram uncertainty. Water Resources Research 47(5)
39. Jafarpour B, McLaughlin DB, et al (2009) Reservoir characterization with the discrete cosine transform. SPE Journal 14(01):182–201
40. Jenatton R, Obozinski G, Bach F (2010) Structured sparse principal component analysis. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pp 366–373
41. Jolliffe IT (1986) Principal component analysis and factor analysis. In: Principal component analysis, Springer, pp 115–128
42. Kandel ER, Schwartz JH, Jessell TM, Siegelbaum SA, Hudspeth AJ, et al (2000) Principles of neural science, vol 4. McGraw-Hill New York
43. Khaninezhad MM, Jafarpour B (2014) Prior model identification during subsurface flow data integration with adaptive sparse representation techniques. Computational Geosciences 18(1):3–16

44. Khaninezhad MM, Jafarpour B, Li L (2012) Sparse geologic dictionaries for subsurface flow model calibration: Part i. inversion formulation. Advances in Water Resources 39:106–121
45. Khaninezhad MM, Jafarpour B, Li L (2012) Sparse geologic dictionaries for subsurface flow model calibration: Part ii. robustness to uncertainty. Advances in water resources 39:122–136
46. Khodabakhshi M, Jafarpour B (2013) A Bayesian mixture-modeling approach for flow-conditioned multiple-point statistical facies simulation from uncertain training images. Water Resources Research 49(1):328–342
47. Kitanidis PK (1997) Introduction to geostatistics: applications in hydrogeology. Cambridge University Press
48. Klema V, Laub A (1980) The singular value decomposition: Its computation and some applications. IEEE Transactions on automatic control 25(2):164–176
49. Landis EM (1934) Capillary pressure and capillary permeability. Physiological Reviews 14(3):404–481
50. Lee J, Kitanidis P (2013) Bayesian inversion with total variation prior for discrete geologic structure identification. Water Resources Research 49(11):7658–7669
51. Li L, Jafarpour B (2010) A sparse Bayesian framework for conditioning uncertain geologic models to nonlinear flow measurements. Advances in Water Resources 33(9):1024–1042
52. Liu X, Kitanidis P (2011) Large-scale inverse modeling with an application in hydraulic tomography. Water Resources Research 47(2)
53. Lochbühler T, Vrugt JA, Sadegh M, Linde N (2015) Summary statistics from training images as prior information in probabilistic inversion. Geophysical Journal International 201(1):157–171
54. Luo J, Wang W, Qi H (2013) Group sparsity and geometry constrained dictionary learning for action recognition from depth maps. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1809–1816
55. Mallat SG (1989) A theory for multiresolution signal decomposition: the wavelet representation. IEEE transactions on pattern analysis and machine intelligence 11(7):674–693
56. Marvasti F, Azghani M, Imani P, Pakrouh P, Heydari SJ, Golmohammadi A, Kazerouni A, Khalili M (2012) Sparse signal processing using iterative method with adaptive thresholding (IMAT). In: Telecommunications (ICT), 2012 19th International Conference on, IEEE, pp 1–6
57. Miller K (1970) Least squares methods for ill-posed problems with a prescribed bound. SIAM Journal on Mathematical Analysis 1(1):52–74
58. Mohimani H, Babaie-Zadeh M, Jutten C (2009) A fast approach for overcomplete sparse decomposition based on smoothed l^0 norm. IEEE Transactions on Signal Processing 57(1):289–301
59. Mueller JL, Siltanen S (2012) Linear and nonlinear inverse problems with practical applications. SIAM
60. Murray CD, Dermott SF (1999) Solar system dynamics. Cambridge university press
61. Needell D, Tropp JA (2009) CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. Applied and Computational Harmonic Analysis 26(3):301–321
62. Oliver DS, Chen Y (2011) Recent progress on reservoir history matching: a review. Computational Geosciences 15(1):185–221
63. Oliver DS, Reynolds AC, Liu N (2008) Inverse theory for petroleum reservoir characterization and history matching. Cambridge University Press
64. Patankar S (1980) Numerical heat transfer and fluid flow. CRC press
65. Peterson AF, Ray SL, Mittra R, of Electrical I, Engineers E (1998) Computational methods for electromagnetics. IEEE press New York
66. Resmerita E (2005) Regularization of ill-posed problems in Banach spaces: convergence rates. Inverse Problems 21(4):1303
67. Riva M, Panzeri M, Guadagnini A, Neuman SP (2011) Role of model selection criteria in geostatistical inverse estimation of statistical data-and model-parameters. Water Resources Research 47(7)
68. Rousset M, Durlofsky L (2014) Optimization-based framework for geological scenario determination using parameterized training images. In: ECMOR XIV-14th European Conference on the Mathematics of Oil Recovery

69. Rudin LI, Osher S, Fatemi E (1992) Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena 60(1–4):259–268
70. Sarma P, Durlofsky LJ, Aziz K (2008) Kernel principal component analysis for efficient, differentiable parameterization of multipoint geostatistics. Mathematical Geosciences 40(1): 3–32
71. Shawe-Taylor J, Cristianini N (2004) Kernel methods for pattern analysis. Cambridge university press
72. Shirangi MG (2014) History matching production data and uncertainty assessment with an efficient TSVD parameterization algorithm. Journal of Petroleum Science and Engineering 113:54–71
73. Shirangi MG, Durlofsky LJ (2016) A general method to select representative models for decision making and optimization under uncertainty. Computers & Geosciences 96:109–123
74. Snieder R (1998) The role of nonlinearity in inverse problems. Inverse Problems 14(3):387
75. Strebelle S (2002) Conditional simulation of complex geological structures using multiple-point statistics. Mathematical Geology 34(1):1–21
76. Suzuki S, Caers JK, et al (2006) History matching with an uncertain geological scenario. In: SPE Annual Technical Conference and Exhibition, Society of Petroleum Engineers
77. Talukder KH, Harada K (2010) Haar wavelet based approach for image compression and quality assessment of compressed image. arXiv preprint arXiv:10104084
78. Tarantola A (2005) Inverse problem theory and methods for model parameter estimation. SIAM
79. Tarantola A, Valette B (1982) Generalized nonlinear inverse problems solved using the least squares criterion. Reviews of Geophysics 20(2):219–232
80. Taubman D, Marcellin M (2012) JPEG2000 image compression fundamentals, standards and practice: image compression fundamentals, standards and practice, vol 642. Springer Science & Business Media
81. Tikhonov A, Arsenin VY (1979) Methods of solving incorrect problems
82. Tosic I, Frossard P (2011) Dictionary learning. IEEE Signal Processing Magazine 28(2):27–38
83. Tropp JA, Gilbert AC (2007) Signal recovery from random measurements via orthogonal matching pursuit. IEEE Transactions on information theory 53(12):4655–4666
84. Vo HX, Durlofsky LJ (2014) A new differentiable parameterization based on principal component analysis for the low-dimensional representation of complex geological models. Mathematical Geosciences 46(7):775–813
85. Vogel CR (2002) Computational methods for inverse problems. SIAM
86. Vrugt JA, Stauffer PH, Wöhling T, Robinson BA, Vesselinov VV (2008) Inverse modeling of subsurface flow and transport properties: A review with new developments. Vadose Zone Journal 7(2):843–864
87. Yeh WWG (1986) Review of parameter identification procedures in groundwater hydrology: The inverse problem. Water Resources Research 22(2):95–108
88. Zhou H, Gómez-Hernández JJ, Li L (2012) A pattern-search-based inverse method. Water Resources Research 48(3)
89. Zhou H, Gómez-Hernández JJ, Li L (2014) Inverse methods in hydrogeology: Evolution and recent trends. Advances in Water Resources 63:22–37
90. Zimmerman D, Marsily Gd, Gotway CA, Marietta MG, Axness CL, Beauheim RL, Bras RL, Carrera J, Dagan G, Davies PB, et al (1998) A comparison of seven geostatistically based inverse approaches to estimate transmissivities for modeling advective transport by groundwater flow. Water Resources Research 34(6):1373–1413