



# Novel iterative ensemble smoothers derived from a class of generalized cost functions

Xiaodong Luo<sup>1</sup>

Received: 22 May 2020 / Accepted: 9 February 2021  
© The Author(s) 2021

## Abstract

Iterative ensemble smoothers (IES) are among the state-of-the-art approaches to solving history matching problems. From an optimization-theoretic point of view, these algorithms can be derived by solving certain stochastic nonlinear-least-squares problems. In a broader picture, history matching is essentially an inverse problem, which is often ill-posed and may not possess a unique solution. To mitigate the ill-posedness, in the course of solving an inverse problem, prior knowledge and domain experience are often incorporated, as a regularization term, into a suitable cost function within a respective optimization problem. Whereas in the inverse theory there is a rich class of inversion algorithms resulting from various choices of regularized cost functions, there are few ensemble data assimilation algorithms (including IES) which in their practical uses are implemented in a form beyond nonlinear-least-squares. This work aims to narrow this noticed gap. Specifically, we consider a class of more generalized cost functions, and establish a unified formula that can be used to construct a corresponding group of novel ensemble data assimilation algorithms, called generalized IES (GIES), in a principled and systematic way. For demonstration, we choose a subset (up to 30+) of the GIES algorithms derived from the unified formula, and apply them to two history matching problems. Experiment results indicate that many of the tested GIES algorithms exhibit superior performance to that of an original IES developed in a previous work, showcasing the potential benefit of designing new ensemble data assimilation algorithms through the proposed framework.

**Keywords** Inverse problem · Data assimilation · History matching · Generalized iterative ensemble smoothers · Mixture of  $\ell_p$  norms

## 1 Introduction

A typical inverse problem involves finding one (or multiple) set(s) of model variables  $\mathbf{m}$ , as inputs to a numerical forward simulator  $\mathbf{g}$ , in such a way that the generated outputs  $\mathbf{g}(\mathbf{m})$  from the simulator can match a collection of observed data  $\mathbf{d}^o$ , or other relevant quantities of interest (e.g., a probability density function conditioned on  $\mathbf{d}^o$ ), to a good extent. Inverse theory is one of the important areas in various science and engineering disciplines, and finds wide applications in different areas, such as system identification, signal/image processing, computer vision, machine learning, meteorology, oceanography, geophysics, petroleum engineering, to name but a few. Depending on the field, inverse theory may have diverse names. For

instance, it is known as estimation theory in statistical signal processing [1], or data assimilation in meteorology and oceanography [2, 3], or history matching in reservoir engineering [4]. In the current work, both the terms “data assimilation” and “history matching” will be adopted and used interchangeably.

To find a solution to an inverse problem, there are two major schools of methods, from the Bayesian and optimization-theoretic perspectives, respectively. A Bayesian approach focuses on updating a prior probability density function (PDF)  $p(\mathbf{m})$  of model variables to the posterior one  $p(\mathbf{m}|\mathbf{d}^o)$  based on Bayes’ theorem, whereas the forward simulator (together with the PDF of observation noise) is used to calculate the likelihood  $p(\mathbf{d}^o|\mathbf{m})$  that the observations  $\mathbf{d}^o$  are caused by a specific set of model variables  $\mathbf{m}$  [5]. After the posterior PDF is obtained, point estimates, such as the corresponding mean, median or mode, can be generated accordingly.

In a conventional optimization-based inversion algorithm [6], one first defines a certain cost function  $C(\mathbf{m}; \mathbf{d}^o, \mathbf{m}_0)$

✉ Xiaodong Luo  
xluo@norce-research.no

<sup>1</sup> Norwegian Research Centre (NORCE), 5008 Bergen, Norway

that describes the discrepancies between the observations  $\mathbf{d}^o$  and the outputs  $\mathbf{g}(\mathbf{m})$  of the forward simulator, where  $\mathbf{m}_0$  represents an initial guess of the unknown model variables<sup>1</sup>. The solution to the inverse problem then boils down to finding one (or multiple) set(s) of model variables  $\mathbf{m}$  that approximately minimize the cost function  $C(\mathbf{m}; \mathbf{d}^o, \mathbf{m}_0)$ . Often, inverse problems are ill-posed, e.g., in the sense that there can exist an infinite number of models  $\mathbf{m}$  that match the observations  $\mathbf{d}^o$  equally well. To mitigate the ill-posedness, it is customary to introduce a certain regularization term to the cost function, such that the regularized cost function is typically in the form of  $C(\mathbf{m}; \mathbf{d}^o, \mathbf{m}_0) = \mathcal{D}(\mathbf{d}^o - \mathbf{g}(\mathbf{m})) + \gamma \mathcal{R}(\mathbf{m} - \mathbf{m}_0)$ , where the operator  $\mathcal{D}$  defines the discrepancies (or mismatch) between the observations  $\mathbf{d}^o$  and the simulated data  $\mathbf{g}(\mathbf{m})$  of the forward simulator, the operator  $\mathcal{R}$  specifies the amount of deviation of  $\mathbf{m}$  from the initial guess  $\mathbf{m}_0$ , and  $\gamma$  is a regularization parameter associated with  $\mathcal{R}$ . As will be shown later, one can slightly extend the form of the cost function  $C(\mathbf{m}; \mathbf{d}^o, \mathbf{m}_0)$ , by allowing  $\mathcal{D}$  and/or  $\mathcal{R}$  to act on transformed data and/or model variables. For ease of reference, hereafter we call  $\mathcal{D}(\mathbf{d}^o - \mathbf{g}(\mathbf{m}))$  and  $\gamma \mathcal{R}(\mathbf{m} - \mathbf{m}_0)$  (or simply  $\mathcal{R}(\mathbf{m} - \mathbf{m}_0)$ ) the data mismatch and regularization terms, respectively.

Under certain circumstances, connections between Bayesian and optimization-based inversion approaches can be established. For instance, if one lets the prior PDF  $p(\mathbf{m}) \propto \exp(-\gamma \mathcal{R}(\mathbf{m} - \mathbf{m}_0))$ , and the likelihood function  $p(\mathbf{d}^o | \mathbf{m}) \propto \exp(-\mathcal{D}(\mathbf{d}^o - \mathbf{g}(\mathbf{m})))$ , then according to Bayes' theorem, the posterior PDF  $p(\mathbf{m} | \mathbf{d}^o) \propto \exp(-\mathcal{D}(\mathbf{d}^o - \mathbf{g}(\mathbf{m})) - \gamma \mathcal{R}(\mathbf{m} - \mathbf{m}_0))$ . Hence, an inverse-problem solution obtained by minimizing the cost function  $\mathcal{D}(\mathbf{d}^o - \mathbf{g}(\mathbf{m})) + \gamma \mathcal{R}(\mathbf{m} - \mathbf{m}_0)$  is equivalent to a maximum a posteriori (MAP) estimate of the PDF  $p(\mathbf{m} | \mathbf{d}^o)$  derived from the Bayesian approach.

Although theoretically sound, computational issues may arise in applications of the conventional Bayesian and optimization-based inversion approaches to practical inverse problems. For a Bayesian approach, the involved PDFs are often replaced by certain Monte Carlo approximations, which typically require to adopt a large number of samples in order to achieve a decent approximation accuracy, and may thus become prohibitively expensive in large-scale problems, where even running a single forward simulation would require substantial computational resources. In contrast, a conventional optimization-based inversion approach tends to be computationally more efficient than a Bayesian approach, as it requires only a few forward simulations per iteration step. However, in many

cases, solving the optimization problem itself may become more complicated. For instance, if a gradient-based local optimization method is adopted to solve the optimization problem, the exact gradient may not always be available, e.g., in case that the forward simulator is a black-box system provided by a third party. In addition, aiming to obtain a single point estimate, an optimization-based inversion approach would not be able to quantify the uncertainties associated with the estimated model variables in general.

Recent years have witnessed the developments of new algorithms that are able to mitigate some of the noticed computational issues in Bayesian and/or optimization-based inversion approaches, among which we focus here on iterative ensemble smoothers (IES) that are among the state-of-the-art approaches to large-scale history matching problems. In this regard, different forms of IES are developed from either a Bayesian or an optimization-theoretic point of view, for example, see, [7–15].

Although in principle one can iteratively update the posterior PDF within the framework of IES, in such a way that the prior and final posterior PDF can satisfy Bayes' theorem [10, 13, 15], in practical implementations it is rare that PDFs are explicitly involved in numerical computations. Instead, the more common choice is to adopt a relatively small ensemble of model variables to represent the whole space that the model variables may span, and the iteration process aims to update the ensemble of model variables under a certain criterion, derived from either the Bayesian or the optimization-theoretic perspective. For a Bayesian approach, e.g., the ensemble smoother with multiple data assimilation (ES-MDA) [10], such an ensemble can be treated as an Monte Carlo approximation to the corresponding PDF, but with a relatively small ensemble size to reduce the computational cost. The iteration process is then designed with the goal that the Monte Carlo approximations of PDFs (approximately) satisfy Bayes' theorem.

Meanwhile, for an optimization-based approach, e.g., [7, 11, 13, 14], the ensemble of model variables leads to an ensemble of cost functions, and the objective is to update the ensemble of model variables to minimize either each individual cost function [8, 11], or the average of the cost functions [13, 14]. Compared to the conventional optimization-based inversion approaches, ensemble-based methods are non-intrusive, and do not require explicit knowledge of the gradients of a cost function with respect to the model variables. As such, ensemble-based methods are derivative-free and applicable to black-box systems, and are more flexible and convenient in terms of algorithm implementations. Moreover, instead of producing a single point estimate as in a conventional optimization-based approach, an ensemble-based method generates an ensemble of estimates, which provides a means of

<sup>1</sup>In the context of atmospheric data assimilation problems, the conventional (either three- or four-dimensional) variational data assimilation methods [3] can be considered to belong to this class of inversion algorithms

uncertainty quantification to some extent. On the other hand, an open problem for ensemble-based methods is that, except for some special cases, in general the final ensemble of estimated model variables may not be considered as samples drawn from the correct posterior PDF, which satisfies Bayes' theorem with respect to the corresponding prior PDF and the likelihood function [16].

For IES developed from either the Bayesian or the optimization-theoretical perspective, a less explored issue is that, in practical applications, the underlying cost functions are typically limited to the forms of nonlinear least squares, in which both the data mismatch and regularization operators, namely  $\mathcal{D}$  and  $\mathcal{R}$ , are some quadratic functions with respect to the simulated data and the model variables, respectively. In contrast, for the conventional optimization-based inversion approaches, there exist richer types of cost functions. For instance, in terms of the regularization operator  $\mathcal{R}$ , except for the conventional quadratic form (known as Tikonov or  $\ell_2$ -norm regularization) as adopted in the IES, other types of regularization methods, such as total variation regularization [17],  $\ell_p$  regularization in which  $p$  may not be equal to 2 [18], mixed-norm regularization [19] etc, are also proposed in the literature to improve the performance of inversion algorithms. However, to the best of our knowledge, in the literature there seem to lack such ensemble-based iterative inversion algorithms that are developed based on cost functions beyond the conventional forms of nonlinear least squares.

This work aims to narrow this noticed gap between the conventional optimization-based inversion approaches and the ensemble-based ones, in terms of the varieties of the underlying cost functions used to establish the corresponding inversion algorithms. The motivation for us to consider other types of cost functions can also be understood from a Bayesian perspective. Indeed, as aforementioned, the Bayesian formulation can have an underlying cost function if we make a connection between the Bayesian and optimization-based approaches under the MAP criterion. For a cost function in the form of nonlinear least squares, it is equivalent for a Bayesian approach to assume that both the prior model variables and the observation noise follow certain Gaussian distributions. In practice, however, it may also be desirable to consider other statistical distributions. One good example in this regard is the Bayesian LASSO regression [20], which equips the regularization term with the  $\ell_1$  norm, corresponding to a Laplacian prior. Another good example is the Bayesian elastic net [21], which admits a mixture of  $\ell_1$  and  $\ell_2$  norms for the regularization term, corresponding to a prior resulting from the combined use of Gaussian and Laplacian distributions.

The rest of this paper is organized as follows: We first present a unified ensemble-based model update

formula, which provides solutions to a class of inverse problems, in the form of generalized minimum average cost (GMAC) problems. Underlying the GMAC problems is a corresponding class of regularized cost functions that generalize and go beyond the form of nonlinear least squares in general. As such, the presented update formula provides a principled and systematic way to derive a group of novel ensemble-based data assimilation algorithms, called (more) generalized iterative ensemble smoothers (GIES) in this work. We show that the update formula of the GIES algorithms bear certain structural similarities to those of the IES in the literature. Consequently, previous experience in implementing and applying the IES algorithms can be naturally transferred to the implementations and applications of the GIES. For demonstration, we compare the performance of a subset (up to 30+) of the GIES algorithms (including one original IES developed in a previous work) in two history matching problems. Experiment results indicate that many of the tested GIES algorithms outperform the original IES in the case studies, manifesting the potential benefit of designing new ensemble data assimilation algorithms under the GMAC framework.

## 2 Generalized iterative ensemble smoothers (GIES)

### 2.1 Original IES derived from a stochastic nonlinear-least-squares problem

Given the following noisy observation system

$$\mathbf{d}^o = \mathbf{g}(\mathbf{m}^{tr}) + \boldsymbol{\epsilon}, \quad (1)$$

where  $\mathbf{d}^o \in \mathbb{R}^d$  stands for a  $d$ -dimensional vector of observations;  $\mathbf{g} : \mathbb{R}^m \rightarrow \mathbb{R}^d$  for a forward simulator that maps an  $m$ -dimensional model vector  $\mathbf{m}^{tr} \in \mathbb{R}^m$  onto the observation space; and  $\boldsymbol{\epsilon} \in \mathbb{R}^d$  for contamination noise following a certain distribution.

In ensemble-based data assimilation, our objective is to estimate a set of possible model vectors  $\mathbf{m}$  that may explain the observations  $\mathbf{d}^o$  to a good extent, given the forward simulator  $\mathbf{g}$  and possibly certain prior knowledge of  $\mathbf{m}$ . Without loss of generality, we assume that the forward simulator  $\mathbf{g}$  is perfect.

Since the forward simulator  $\mathbf{g}$  is nonlinear in general, we employ a certain iterative algorithm to solve the data assimilation problem. To this end, let us assume that at the  $i$ th iteration step, we have an ensemble of  $N_e$  model vectors, denoted by  $\mathcal{M}^i \equiv \{\mathbf{m}_j^i\}_{j=1}^{N_e}$ , as the initial guess, where  $j$  is the index of ensemble members. Following the previous work [13], our objective is to find a new ensemble  $\mathcal{M}^{i+1} \equiv$

$\{\mathbf{m}_j^{i+1}\}_{j=1}^{N_e}$  of model vectors that approximately solves the following minimum-average-cost (MAC) problem, in the form of stochastic nonlinear-least-squares, at the  $(i + 1)$ th iteration step:

$$\arg \min_{\{\mathbf{m}_j^{i+1}\}_{j=1}^{N_e}} \frac{1}{2N_e} \sum_{j=1}^{N_e} \left\{ (\mathbf{d}_j - \mathbf{g}(\mathbf{m}_j^{i+1}))^T \mathbf{C}_d^{-1} (\mathbf{d}_j - \mathbf{g}(\mathbf{m}_j^{i+1})) + \gamma^i (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)^T (\mathbf{C}_m^i)^{-1} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i) \right\}, \quad (2)$$

where  $\mathbf{d}_j$  is a data vector being either the observation vector  $\mathbf{d}^o$  itself or a perturbed version of  $\mathbf{d}^o$ ;  $\mathbf{C}_m^i$  is the sample covariance matrix induced by the ensemble  $\mathcal{M}^i$ , in the form of  $\mathbf{C}_m^i = \mathbf{S}_Z^i (\mathbf{S}_Z^i)^T$ , with the square-root matrix  $\mathbf{S}_Z^i$  being defined later (cf. Eq. (6)); and  $\gamma^i$  is a parameter that is adaptive to the iteration process, following certain preset rules [13].

The main idea in Eq. 2 is to minimize the average of an ensemble of cost functions. Each cost function consists of two parts, namely, the data mismatch term which measures the weighted euclidean distance between simulated and observed data, and the quadratic regularization term which is introduced to mitigate the ill-posedness in the history matching problem, and also to prevent the IES from overfitting the observations.

Through a linearization-based approximation strategy, Eq. 2 can be solved as follows [13]:

$$\mathbf{m}_j^{i+1} = \mathbf{m}_j^i + \mathbf{K}^i (\mathbf{d}_j - \mathbf{g}(\mathbf{m}_j^i)), \quad j = 1, 2, \dots, N_e; \quad (3)$$

$$\begin{aligned} \mathbf{K}^i &\equiv \mathbf{S}_Z^i (\mathbf{S}_g^i)^T (\mathbf{S}_g^i (\mathbf{S}_g^i)^T + \gamma^i \mathbf{C}_d)^{-1} \\ &= \mathbf{S}_Z^i \left[ (\mathbf{S}_g^i)^T \mathbf{C}_d^{-1} \mathbf{S}_g^i + \gamma^i \mathbf{I}_{N_e} \right]^{-1} (\mathbf{S}_g^i)^T \mathbf{C}_d^{-1}; \end{aligned} \quad (4)$$

$$\bar{\mathbf{m}}^i \equiv \frac{1}{N_e} \sum_{j=1}^{N_e} \mathbf{m}_j^i; \quad (5)$$

$$\mathbf{S}_Z^i \equiv \frac{1}{\sqrt{N_e - 1}} [\mathbf{m}_1^i - \bar{\mathbf{m}}^i, \mathbf{m}_2^i - \bar{\mathbf{m}}^i, \dots, \mathbf{m}_{N_e}^i - \bar{\mathbf{m}}^i]; \quad (6)$$

$$\mathbf{S}_g^i \equiv \frac{1}{\sqrt{N_e - 1}} [\mathbf{g}(\mathbf{m}_1^i) - \mathbf{g}(\bar{\mathbf{m}}^i), \mathbf{g}(\mathbf{m}_2^i) - \mathbf{g}(\bar{\mathbf{m}}^i), \dots, \mathbf{g}(\mathbf{m}_{N_e}^i) - \mathbf{g}(\bar{\mathbf{m}}^i)]. \quad (7)$$

For the proof of the matrix identity in Eq. 4, readers are referred to Appendix A.

For notational convenience, the square-root matrices in Eqs. 6 and 7 can be expressed in a unified way as follows:

$$\mathbf{S}_{\mathcal{O}}^i \equiv \frac{1}{\sqrt{N_e - 1}} [\mathcal{O}(\mathbf{m}_1^i) - \mathcal{O}(\bar{\mathbf{m}}^i), \mathcal{O}(\mathbf{m}_2^i) - \mathcal{O}(\bar{\mathbf{m}}^i), \dots, \mathcal{O}(\mathbf{m}_{N_e}^i) - \mathcal{O}(\bar{\mathbf{m}}^i)], \quad (8)$$

where  $\mathcal{O}$  represents a certain operator that maps a reservoir model into another domain. In Eq. 6,  $\mathcal{O}$  is reduced to the identity operator  $\mathcal{I}$ , such that  $\mathcal{I}(\mathbf{m}) = \mathbf{m}$  for a reservoir model  $\mathbf{m}$ . Similarly, in Eq. 7, the operator  $\mathcal{O}$  equals the forward reservoir simulator  $\mathbf{g}$ . The custom in Eq. 8 will be adopted throughout this work.

## 2.2 GIES derived from a class of generalized cost functions

In the sequel, we consider the following generalized MAC (GMAC) problem

$$\arg \min_{\{\mathbf{m}_j^{i+1}\}_{j=1}^{N_e}} \frac{1}{N_e} \sum_{j=1}^{N_e} C_j^{i+1}, \quad (9)$$

$$\begin{aligned} C_j^{i+1} &\equiv \mathcal{D} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T}(\mathbf{g}(\mathbf{m}_j^{i+1})) \right] \\ &\quad + \gamma^i \mathcal{R} \left[ \mathcal{S}(\mathbf{m}_j^{i+1}) - \mathcal{S}(\mathbf{m}_j^i) \right], \end{aligned} \quad (10)$$

with a class of more general cost functions, in comparison to the original MAC problem in Eq. 2. According to Eq. 10, in the GMAC problem, each of the cost functions also consists of two terms, namely, the data mismatch term with respect to the distance metric  $\mathcal{D} : \mathbb{R}^t \rightarrow [0, +\infty)$ , and the regularization term with respect to the regularization operator  $\mathcal{R} : \mathbb{R}^s \rightarrow [0, +\infty)$ .

Within the data mismatch term, a transform operator  $\mathcal{T} : \mathbb{R}^d \rightarrow \mathbb{R}^t$  is introduced to accommodate the possibility that the observed and simulated data may be transformed into another domain, e.g., through discrete wavelet transform [22, 23] or learned dictionary [24], before computing respective data mismatch between them. Similarly, within the regularization term, a transform operator  $\mathcal{S} : \mathbb{R}^m \rightarrow \mathbb{R}^s$  is present to take into account the possibility that regularization may be applied to transformed model variables, rather than directly to the model variables themselves. For example, see [25–27]. In this regard,  $\mathcal{S}$  may also be considered as a re-parametrization strategy, if one assumes that the transform operator  $\mathcal{S}$  is (approximately) invertible, so that in the data mismatch term, one can re-write

$$\mathcal{T}(\mathbf{g}(\mathbf{m}_j^{i+1})) = \mathcal{T} \circ \mathbf{g} \circ \mathcal{S}^{-1}(\mathcal{S}(\mathbf{m}_j^{i+1})),$$

where  $\mathcal{T} \circ \mathbf{g} \circ \mathcal{S}^{-1}$  denotes the composition of the operators  $\mathcal{T}$ ,  $\mathbf{g}$  and  $\mathcal{S}^{-1}$ , and acts as an effective forward simulator for the effective input  $\mathcal{S}(\mathbf{m}_j^{i+1})$ . On the other hand, we note that in Eq. (10), the invertibility of  $\mathcal{S}$  is not formally required.

As a special case, given a dummy vector  $\mathbf{x}$ , if one takes

$$\mathcal{D}(\mathbf{x}) \equiv \frac{1}{2} \mathbf{x}^T \mathbf{C}_d^{-1} \mathbf{x}, \quad (11)$$

$$\mathcal{R}(\mathbf{x}) \equiv \frac{1}{2} \mathbf{x}^T (\mathbf{C}_m^i)^{-1} \mathbf{x}, \quad (12)$$

where  $\mathbf{C}_d$  and  $\mathbf{C}_m^i$  represent the observation error covariance matrix and the sample model error covariance matrix, respectively, as in the original MAC problem (2); and also lets both  $\mathcal{T}$  and  $\mathcal{S}$  be identity operators, then up to a constant factor, the GMAC problem in Eqs. 9 and 10 is reduced to the

stochastic nonlinear-least-squares problem in Eq. 2, which leads to the IES derived in [13], and is also closely related to the cost functions in other variants of the IES, e.g., [7, 10, 14].

In Appendices A and B, we show that the GMAC problem can be approximately solved under different scenarios, depending on whether relevant gradient and/or Hessian information is available or not. When all relevant gradient and/or Hessian information is provided, then the GMAC problem is solved by the formulas in Eqs. 62–66. Otherwise, ensemble-based approximate solutions can be obtained, by combining the formulas in Eqs. 74 and 75 and the approximation strategies in Appendix B.

In the current work, we assume that both the distance metric  $\mathcal{D}$  and the regularization operator  $\mathcal{R}$  are relatively simple (e.g., as in Eqs. 11 and 12), such that the analytic forms of gradient and Hessian of  $\mathcal{D}$  (with respect to  $\mathcal{T}$ ) and  $\mathcal{R}$  (with respect to  $\mathcal{S}$ ) are available.

Under this setting, it can be shown (cf. Appendix A) that an ensemble-based approximate solution to the GMAC problem in Eqs. 9 and 10 is given by

$$\mathbf{m}_j^{i+1} = \mathbf{m}_j^i + \tilde{\mathbf{K}}_j^i \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)]; \quad (13)$$

$$\tilde{\mathbf{K}}_j^i \equiv \mathbf{S}_{\mathcal{T}}^i [\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) + \nu^i \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)]^{-1} (\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i)^T; \quad (14)$$

$$\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) \equiv (\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i)^T \nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\bar{\mathbf{m}}^i)] \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i; \quad (15)$$

$$\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) \equiv (\mathbf{S}_{\mathcal{S}}^i)^T \nabla_{\mathcal{R}}^2 [\mathcal{S}(\bar{\mathbf{m}}^i) - \mathcal{S}(\mathbf{m}_j^i)] \mathbf{S}_{\mathcal{S}}^i, \quad (16)$$

where, given an operator  $\mathcal{O}$  and a dummy vector variable  $\mathbf{x}$  (and a concrete value of  $\mathbf{x}$  at  $\mathbf{x}_0$ ),

$$\begin{aligned} \nabla_{\mathcal{O}}[\mathbf{x}_0] &\equiv \left. \frac{\partial \mathcal{O}}{\partial \mathbf{x}} \right|_{\mathbf{x}_0}; \\ \nabla_{\mathcal{O}}^2[\mathbf{x}_0] &\equiv \left. \frac{\partial^2 \mathcal{O}}{\partial \mathbf{x}^2} \right|_{\mathbf{x}_0}, \end{aligned} \quad (17)$$

and the definitions of the square-root matrices  $\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i$  and  $\mathbf{S}_{\mathcal{S}}^i$  follow the custom in Eq. 8.

If the distance metric  $\mathcal{D}$  and the regularization operator  $\mathcal{R}$  take the form in Eqs. 11 and 12, respectively, and both  $\mathcal{T}$  and  $\mathcal{S}$  are identity operators, then it can be shown (cf. Appendix A) that the update formula, Eqs. 13–16, is reduced to that in the original IES, Eqs. 3–7. For this reason, we call the IES resulting from Eqs. 13–16 the generalized IES (GIES).

Comparing the update formulas of the original IES and the GIES, one can see that they bear certain structural similarities, as is evidenced in the correspondence between the original IES and the GIES (cf. Table 1). In both update formulas, the model changes, in terms of  $\mathbf{m}_j^{i+1} - \mathbf{m}_j^i$ , can be expressed as the product between an “effective”

Kalman-gain-like matrix and an “effective” innovation term. In the GIES, the effective innovation term can be interpreted as a gradient with respect to (w.r.t) the data mismatch (after data transform) of the respective ensemble member.

Of particular interest here is the dependence of the effective Kalman-gain-like matrix on individual ensemble members in the GIES, which is a phenomenon not observed in the original IES. Indeed, as indicated in Table 1, the effective Kalman-gain-like matrix in the original IES is common to all ensemble members, for which the reason is the quadratic forms of the distance metric  $\mathcal{D}$  and the regularization operator  $\mathcal{R}$  adopted in Eqs. 11 and 12, such that the corresponding local Hessian matrices,  $\mathbf{C}_d^{-1}$  and  $\mathbf{C}_m^{-1}$ , respectively, are independent of model variables. In contrast, in the GIES, the distance metric  $\mathcal{D}$  and the regularization operator  $\mathcal{R}$  may not necessarily admit constant Hessian matrices everywhere. Instead, in general, these Hessian matrices may vary for different ensemble members, and thus depend on the iteration path of the ensemble algorithm.

In implementation, the dependence of the Kalman-gain-like matrix on individual ensemble members implies that the GIES would require more computational time and computer memory than the original IES does. Fortunately, since both the projected local Hessian (PLH) matrices,  $\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)$  and  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$ , respectively, are in the dimension of  $N_e \times N_e$ , the additional memory consumption would not be significant when the ensemble size  $N_e$  is not too big. On the other hand, in terms of the computational complexity, the numbers of flops required to evaluate  $\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)$  and  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$  are in the orders of  $t^2 \times N_e$  and  $s^2 \times N_e$ , respectively, where  $t$  and  $s$  are the sizes of transformed data and model, after applying the transform operators  $\mathcal{T}$  and  $\mathcal{S}$ . Based on these observations, there are a few strategies that can be exploited to help reduce the required computational resources. For instance, if one chooses not to perturb the observations, then the PLH  $\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)$  will be the same for all reservoir models. On the other hand, sparse representations of model and/or data, e.g., [22, 24, 26, 27], can also be adopted to render reduced sizes for transformed data and model.

### 2.3 A subclass of the GIES: The $\ell_p^q$ -GIES

For illustration, in this work we consider a subclass of the GIES algorithms, which approximately solves the following GMAC problem

$$\arg \min_{\{\mathbf{m}_j^{i+1}\}_{j=1}^{N_e}} \frac{1}{N_e} \sum_{j=1}^{N_e} C_j^{i+1}; \quad (18)$$



**Table 1** Correspondence between the original IES and the GIES

	Original IES	GIES
Effective Kalman-gain-like matrix	$\mathbf{S}_{\mathcal{I}}^i \left[ \left( \mathbf{S}_{\mathbf{g}}^i \right)^T \mathbf{C}_d^{-1} \mathbf{S}_{\mathbf{g}}^i + \gamma^i \mathbf{I}_{N_e} \right]^{-1} \left( \mathbf{S}_{\mathbf{g}}^i \right)^T$	$\mathbf{S}_{\mathcal{I}}^i \left[ \mathbf{M}_{\mathcal{D}}^i \left( \mathbf{d}_j \right) + \gamma^i \mathbf{M}_{\mathcal{R}}^i \left( \mathbf{m}_j^i \right) \right]^{-1} \left( \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right)^T$
Effective innovation term	$\mathbf{C}_d^{-1} \left( \mathbf{d}_j - \mathbf{g} \left( \mathbf{m}_j^i \right) \right)$	$\nabla_{\mathcal{D}} \left[ \mathcal{T} \left( \mathbf{d}_j \right) - \mathcal{T} \circ \mathbf{g} \left( \mathbf{m}_j^i \right) \right]$
Local Hessian w.r.t the data mismatch term	$\mathbf{C}_d^{-1}$	$\nabla_{\mathcal{D}}^2 \left[ \mathcal{T} \left( \mathbf{d}_j \right) - \mathcal{T} \circ \mathbf{g} \left( \bar{\mathbf{m}}^i \right) \right]$
Local Hessian w.r.t the regularization term	$\left( \mathbf{C}_m^i \right)^{-1}$	$\nabla_{\mathcal{R}}^2 \left[ \mathcal{S} \left( \bar{\mathbf{m}}^i \right) - \mathcal{S} \left( \mathbf{m}_j^i \right) \right]$
Projected local Hessian w.r.t the data mismatch term	$\left( \mathbf{S}_{\mathbf{g}}^i \right)^T \mathbf{C}_d^{-1} \mathbf{S}_{\mathbf{g}}^i$	$\mathbf{M}_{\mathcal{D}}^i \left( \mathbf{d}_j \right) \equiv \left( \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right)^T \nabla_{\mathcal{D}}^2 \left[ \mathcal{T} \left( \mathbf{d}_j \right) - \mathcal{T} \circ \mathbf{g} \left( \bar{\mathbf{m}}^i \right) \right] \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i$
Projected local Hessian w.r.t the regularization term	$\mathbf{I}_{N_e}$	$\mathbf{M}_{\mathcal{R}}^i \left( \mathbf{m}_j^i \right) \equiv \left( \mathbf{S}_{\mathcal{S}}^i \right)^T \nabla_{\mathcal{R}}^2 \left[ \mathcal{S} \left( \bar{\mathbf{m}}^i \right) - \mathcal{S} \left( \mathbf{m}_j^i \right) \right] \mathbf{S}_{\mathcal{S}}^i$
Square-root matrix in the (transformed) data space	$\mathbf{S}_{\mathbf{g}}^i$	$\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i$
Square-root matrix in the (transformed) model space	$\mathbf{S}_{\mathcal{I}}^i$	$\mathbf{S}_{\mathcal{S}}^i$

$$\mathbf{C}_j^{i+1} = \frac{1}{2} \left( \mathbf{d}_j - \mathbf{g} \left( \mathbf{m}_j^{i+1} \right) \right)^T \mathbf{C}_d^{-1} \left( \mathbf{d}_j - \mathbf{g} \left( \mathbf{m}_j^{i+1} \right) \right) + \gamma^i \mathcal{R} \left[ \mathcal{S} \left( \mathbf{m}_j^{i+1} \right) - \mathcal{S} \left( \mathbf{m}_j^i \right) \right]; \quad (19)$$

$$\mathcal{R} \left[ \mathcal{S} \left( \mathbf{m}_j^{i+1} \right) - \mathcal{S} \left( \mathbf{m}_j^i \right) \right] \equiv \frac{1}{2} \sum_{k=1}^{K_{mix}} w_k \|\mathbf{B}_k \left( \mathcal{S}_k \left( \mathbf{m}_j^{i+1} \right) - \mathcal{S}_k \left( \mathbf{m}_j^i \right) \right)\|_{p_k}^{q_k}; w_k \geq 0, \quad (20)$$

where for a  $N_b \times m_x$  matrix  $\mathbf{B}$  and a  $m_x$ -dimensional vector  $\mathbf{x} = [x_1, x_2, \dots, x_{m_x}]^T$ , we have the  $\ell_p^q$  metric  $\|\mathbf{B}\mathbf{x}\|_p^q$  defined as

$$\|\mathbf{B}\mathbf{x}\|_p^q = \left( \sum_{e=1}^{N_b} |(\mathbf{B}\mathbf{x})_e|^p \right)^{q/p};$$

$$(\mathbf{B}\mathbf{x})_e \equiv \sum_{f=1}^{m_x} B_{e,f} x_f.$$

Here,  $|\bullet|$  is the operator taking absolute values element-wise, e.g.,  $|\mathbf{x}| = [|x_1|, |x_2|, \dots]^T$ ;  $(\mathbf{B}\mathbf{x})_e$  stands for the  $e$ th element of the vector  $\mathbf{B}\mathbf{x}$ ,  $x_f$  for the  $f$ th element of the vector  $\mathbf{x}$ , and  $B_{e,f}$  for the element of the matrix  $\mathbf{B}$  on the  $e$ th row and the  $f$ th column.

In this GMAC problem, the data mismatch term is in the conventional form of nonlinear least-squares, which corresponds to the distance metric  $\mathcal{D}$  in Eq. 11, with  $\mathcal{T}$

being an identity transform operator. On the other hand, the regularization term consists of a mixture of distance metrics  $\|\mathbf{B}_k \left( \mathcal{S}_k \left( \mathbf{m}_j^{i+1} \right) - \mathcal{S}_k \left( \mathbf{m}_j^i \right) \right)\|_{p_k}^{q_k}$ , weighted by non-negative coefficients  $w_k$ . For brevity, we refer to  $\|\mathbf{B}_k \mathbf{x}\|_{p_k}^{q_k}$  as the  $\ell_{p_k}^{q_k}$  metric w.r.t  $\mathbf{B}_k \mathbf{x}$ , and call the resulting subclass of inversion algorithms  $\ell_p^q$ -GIES. Although not considered in the current work, the derivation to be presented below can be similarly applied to establish variants of the  $\ell_p^q$ -GIES, for which the corresponding distance metric  $\mathcal{D}$  also consists of a mixture of different operators.

According to the update formula of the GIES, Eqs. 13–16, we have

$$\nabla_{\mathcal{D}} \left[ \mathcal{T} \left( \mathbf{d}_j \right) - \mathcal{T} \circ \mathbf{g} \left( \mathbf{m}_j^i \right) \right] = \mathbf{C}_d^{-1} \left( \mathbf{d}_j - \mathbf{g} \left( \mathbf{m}_j^i \right) \right); \quad (21)$$

$$\mathbf{M}_{\mathcal{D}}^i \left( \mathbf{d}_j \right) = \left( \mathbf{S}_{\mathbf{g}}^i \right)^T \mathbf{C}_d^{-1} \mathbf{S}_{\mathbf{g}}^i; \quad (22)$$

$$\mathbf{m}_j^{i+1} = \mathbf{m}_j^i + \mathbf{S}_{\mathcal{I}}^i \left[ \left( \mathbf{S}_{\mathbf{g}}^i \right)^T \mathbf{C}_d^{-1} \mathbf{S}_{\mathbf{g}}^i + \gamma^i \mathbf{M}_{\mathcal{R}}^i \left( \mathbf{m}_j^i \right) \right]^{-1}$$

$$\left( \mathbf{S}_{\mathbf{g}}^i \right)^T \mathbf{C}_d^{-1} \left( \mathbf{d}_j - \mathbf{g} \left( \mathbf{m}_j^i \right) \right).$$

By applying the matrix identity (49) of Appendix A to Eq. 23, one obtains the following alternative update formula

$$\mathbf{m}_j^{i+1} = \mathbf{m}_j^i + \mathbf{S}_{\mathcal{I}}^i \left( \mathbf{M}_{\mathcal{R}}^i \left( \mathbf{m}_j^i \right) \right)^{-1} \left( \tilde{\mathbf{S}}_{\mathbf{g}}^i \right)^T \left[ \tilde{\mathbf{S}}_{\mathbf{g}}^i \left( \mathbf{M}_{\mathcal{R}}^i \left( \mathbf{m}_j^i \right) \right)^{-1} \left( \tilde{\mathbf{S}}_{\mathbf{g}}^i \right)^T + \gamma^i \mathbf{I} \right]^{-1} \left( \tilde{\mathbf{d}}_j - \tilde{\mathbf{g}} \left( \mathbf{m}_j^i \right) \right); \quad (24)$$

$$\tilde{\mathbf{S}}_{\mathbf{g}}^i \equiv \mathbf{C}_d^{-1/2} \mathbf{S}_{\mathbf{g}}^i; \tilde{\mathbf{d}}_j \equiv \mathbf{C}_d^{-1/2} \mathbf{d}_j; \tilde{\mathbf{g}} \left( \mathbf{m}_j^i \right) \equiv \mathbf{C}_d^{-1/2} \mathbf{g} \left( \mathbf{m}_j^i \right). \quad (25)$$

The reason for us to present the alternative update formula (24) is that, when combining with a truncated

eigenvalue decomposition, adopting (24) tends to achieve better history matching performance than a straightforward implementation of Eq. 23, in some of the experiments conducted in the current work (not reported here for brevity).

Next, we explain how to evaluate the PLH  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$ . By treating  $\mathcal{R}_k(\bullet) \equiv \|\mathbf{B}_k(\bullet)\|_{p_k}^{q_k}$  as individual operators in the mixture (20), a combination of the analytic results in Appendices C and D leads to the following results for the PLH  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$ , namely,

$$\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) = \frac{1}{2} \sum_{k=1}^{K_{mix}} w_k (\mathbf{S}_{\mathcal{S}_k}^i)^T \left\{ \nabla_{\|\mathbf{B}_k(\bullet)\|_{p_k}^{q_k}}^2 \left[ \mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i) \right] \right\} (\mathbf{S}_{\mathcal{S}_k}^i); \tag{26}$$

$$\begin{aligned} \nabla_{\|\mathbf{B}_k(\bullet)\|_{p_k}^{q_k}}^2 \left[ \mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i) \right] &= q_k (q_k - p_k) \|\mathbf{B}_k(\mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i))\|_{p_k}^{q_k - 2p_k} \mathbf{B}_k^T \mathbf{a}_k \mathbf{a}_k^T \mathbf{B}_k \\ &\quad + q_k (p_k - 1) \|\mathbf{B}_k(\mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i))\|_{p_k}^{q_k - p_k} (\mathbf{B}_k \odot \mathbf{C}_k)^T (\mathbf{B}_k \odot \mathbf{C}_k); \end{aligned} \tag{27}$$

$$\mathbf{a}_k \equiv \left[ \left| \mathbf{B}_k(\mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i)) \right|^{\wedge(p_k - 2)} \right] \odot (\mathbf{B}_k(\mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i))); \tag{28}$$

$$\mathbf{C}_k \equiv \left[ \left| \mathbf{B}_k(\mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i)) \right|^{\wedge(p_k/2 - 1)} \right] \mathbf{1}^T. \tag{29}$$

In Eqs. 28 and 29, the operator  $\wedge$  raises all elements of a vector to a certain power, e.g., given a vector  $\mathbf{x} = [x_1, x_2, \dots]^T$ , one has  $\mathbf{x}^{\wedge p} = [x_1^p, x_2^p, \dots]^T$ ; the operator  $\odot$  stands for element-wise product (i.e., Hadamard product); and  $\mathbf{1}$  for a vector whose elements are all equal to 1.

### 3 A comparison study on $\ell_p^q$ -GIES algorithms applied to a history matching problem

The focus of this section is to show the abundance of the GIES algorithms, illustrated by applying a set of  $\ell_p^q$ -GIES algorithms derived from the previous section to a channelized reservoir characterization problem below. Another set of  $\ell_p^q$ -GIES algorithms is also applied to history matching in a 5-spots problem. To limit the length of the main text, however, numerical results with respect to the 5-spots problem are presented in Appendix E.

#### 3.1 Experiment settings of the history matching problem

##### 3.1.1 The reservoir model

The case study considers a 2D channelized reservoir model in the dimension of  $45 \times 45$ , with oil/water two-phase flow, as described in the previous work [28]. Figure 1 shows the geological structure of the permeability map in the reference model. The porosity of the reference model is spatially constant, and is set to 0.1. On the other hand, the

reservoir has isotropic permeability (PERM), whose values is set to 10000 md within the channel, and to 500 md in the background. The reservoir has 8 injectors (labeled as  $I1, I2$ , etc) and 8 producers (labeled as  $P1, P2$ , etc), whose positions are indicated in Figure 1. The injectors are constrained by injection rates, and the producers by bottom-hole pressures.

The reference model in Fig. 1 is used to generate production data every 190 days, for a total period of 3800 days. The production data consists of well oil/water production rates (WOPR/WWPR) from 8 producers, and well bottom-hole pressures (WBHP) from 8 injectors. In the case study, we divide the production data into two parts. We use the production data (with observation noise) within the first 1900 days for history matching through the  $\ell_p^q$ -GIES algorithms, while reserving the production data (without observation noise) within the second 1900 days for performance validation. The numbers of production data in both time intervals are equal to 240. The observed production data within the history matching period are contaminated by certain zero-mean Gaussian noise. For WOPR/WWPR data, the standard deviations (STDs) of observation noise are  $0.2236 \text{ m}^3/d$ , whereas for WBHP data, their STDs are 0.2646 bar.

In the current case study, permeability values (in the unit of md) are the unknown parameters to be estimated through certain  $\ell_p^q$ -GIES algorithms. An initial ensemble of 100 PERM maps is generated by SNESIM [29], following the procedure explained in [28]. For illustration, Fig. 2 shows the mean (top left) and STD (top right) maps of the initial ensemble, as well as two sample reservoir models from the initial ensemble.

### 3.1.2 The chosen $\ell_p^q$ -GIES algorithms for comparison

In the experiment, the orders of  $p_k$  and  $q_k$  in Eq. 20 are set to either (case  $\ell_1^2$ )  $p_k = 1$  and  $q_k = 2$ ; or (case  $\ell_2^2$ )

$p_k = q_k = 2$ , while the settings of the matrices  $\mathbf{B}_k$  and the transform operators  $\mathcal{S}_k$  will be specified later. According to Eqs. (27) – (29), the corresponding Hessian matrices are then given by

$$\begin{aligned} \nabla_{\|\mathbf{B}_k(\bullet)\|_1^2}^2 \left[ \mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i) \right] &= 2\mathbf{B}_k^T \operatorname{s\,gn} \left( \mathbf{B}_k \left( \mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i) \right) \right) \operatorname{s\,gn} \left( \mathbf{B}_k \left( \mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i) \right) \right)^T \mathbf{B}_k \\ &= 2\mathbf{B}_k^T \operatorname{s\,gn} \left( \mathbf{B}_k \left( \mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i) \right) \left( \mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i) \right)^T \mathbf{B}_k^T \right) \mathbf{B}_k; \end{aligned} \tag{30}$$

$$\nabla_{\|\mathbf{B}_k(\bullet)\|_2^2}^2 \left[ \mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i) \right] = 2\mathbf{B}_k^T \mathbf{B}_k. \tag{31}$$

In Eq. 30,  $\operatorname{s\,gn}(\mathbf{x}) = \mathbf{x} / |\mathbf{x}|$  represents the sign function applying element-wise to its argument  $\mathbf{x}$ , with  $./$  denoting the element-wise division operator. To derive the second line of Eq. 30 from the first one, we have exploited the fact that  $\operatorname{s\,gn}(\mathbf{x}) \operatorname{s\,gn}(\mathbf{x}^T) = \operatorname{s\,gn}(\mathbf{x}\mathbf{x}^T)$ . Another remark is that  $\nabla_{\|\mathbf{B}_k(\bullet)\|_1^2}^2 \left[ \mathcal{S}_k(\bar{\mathbf{m}}^i) - \mathcal{S}_k(\mathbf{m}_j^i) \right] = \mathbf{0}$  almost everywhere, as shown in Appendix C, and this is the reason that we do not consider the  $\ell_1^1$  metric here.

Regarding the transform operator  $\mathcal{S}_k$ , we consider three possible scenarios: (S-I)  $\mathcal{S}_k$  is the identity operator  $\mathcal{I}$ ; (S-II)  $\mathcal{S}_k$  calculates the first-order variations of the permeability map of a reservoir model; and (S-III)  $\mathcal{S}_k$  generates the histogram of the permeability map of a reservoir model, and converts the histogram values into information entropies.

For distinction later, we denote the identity operator in S-I by  $\mathcal{I}$ , such that  $\mathcal{I}(\mathbf{m}) = \mathbf{m}$ . Here we only consider case  $\ell_2^2$ , as this leads to the original IES, and serves as the base case for our comparison study later. Indeed, with  $p_k = q_k = 2$ , the corresponding Hessian is given by Eq. 31. If we let  $\mathbf{B}_k = (\mathbf{S}_{\mathcal{I}}^i)^{-T}$  ( $\mathbf{S}_{\mathcal{I}}$  is the square-root matrix defined in Eq. 6), such that  $\mathbf{B}_k \mathbf{B}_k^T = (\mathbf{S}_{\mathcal{I}}^i (\mathbf{S}_{\mathcal{I}}^i)^T)^{-1} = (\mathbf{C}_m^i)^{-1}$  in Eq. 31; and also choose  $\|\mathbf{B}_k(\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)\|_2^2$  as the only metric in the regularization term (and the associated weight  $w_1 = 1$ ), then according to Eq. 26, the PLH, denoted by  $\mathbf{M}_{\ell_2^2}^i(\mathbf{m}_j^i)$ , equals  $\frac{1}{2} \times 2 (\mathbf{S}_{\mathcal{I}}^i)^T \mathbf{B}_k \mathbf{B}_k^T \mathbf{S}_{\mathcal{I}}^i = \mathbf{I}_{N_c}$ . As a result, the update formula of the  $\ell_p^q$ -GIES, Eq. 23, reduces to that of the original IES, Eqs. 3–4. Note that the subscript of the PLH  $\mathbf{M}_{\ell_2^2}^i(\mathbf{m}_j^i)$ ,  $\ell_2^2$ , combines the information of the transform operator (“ $\mathcal{I}$ ” for identity here) and the  $\ell_p^q$  metric.

The same custom will be adopted for other PLHs later on.

In S-II, the transform operator calculates the first-order spatial variations of permeability maps. For this reason, we denote this operator by  $\mathcal{S}_V$  hereafter. In the current case study, a permeability map can be represented by a  $45 \times 45$  matrix. Let  $\mathbf{P}$  stand for such a matrix, then one can compute the first-order variations of  $\mathbf{P}$  along both the horizontal and vertical directions, denoted by  $\Delta\mathbf{P}_h$  and  $\Delta\mathbf{P}_v$ , respectively. Specifically, suppose that  $\mathbf{P}$  consists of a set of column vectors  $\mathbf{c}_k$ , for  $k = 1, 2, \dots, 45$ , and a set of row vectors  $\mathbf{r}_\ell$  for  $\ell = 1, 2, \dots, 45$ , such that  $\mathbf{P} = [\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_{45}] = [(\mathbf{r}_1)^T, (\mathbf{r}_2)^T, \dots, (\mathbf{r}_{45})^T]^T$ , then we define

$$\Delta\mathbf{P}_h \equiv [\mathbf{c}_2 - \mathbf{c}_1, \mathbf{c}_3 - \mathbf{c}_2, \dots, \mathbf{c}_{45} - \mathbf{c}_{44}]; \tag{32}$$

$$\Delta\mathbf{P}_v \equiv [(\mathbf{r}_2)^T - (\mathbf{r}_1)^T, (\mathbf{r}_3)^T - (\mathbf{r}_2)^T, \dots, (\mathbf{r}_{45})^T - (\mathbf{r}_{44})^T]^T \tag{33}$$

For illustration, Fig. 3a shows the permeability map of sample realization 1 from the initial ensemble, and Fig. 3b and c indicate the first-order spatial variations of the permeability map, along horizontal and vertical directions, respectively. Clearly, in this case, non-zero variation values are able to capture the positions of boundaries between the channel and the background regions.

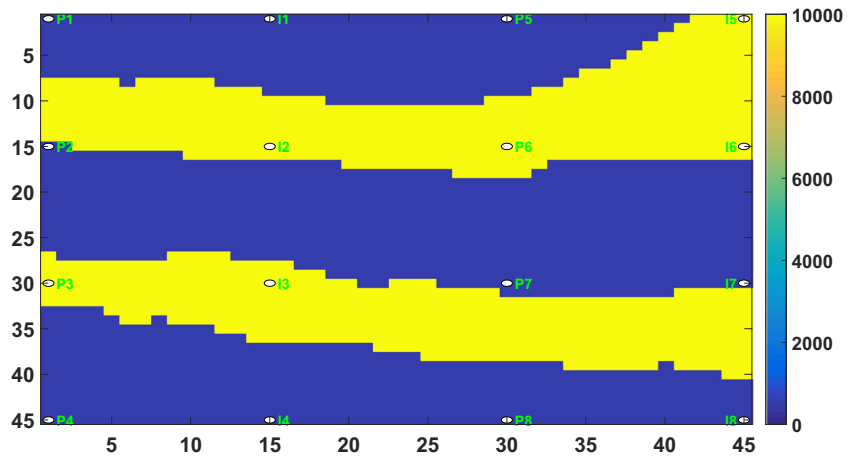
In addition, the transform operator  $\mathcal{S}_V$  further augments the column vectors of both  $\Delta\mathbf{P}_h$  and  $\Delta\mathbf{P}_v$  into a super column vector. As such, given two reservoir models, say  $\mathbf{m}_1$  and  $\mathbf{m}_2$ , the difference  $\mathcal{S}_V(\mathbf{m}_1) - \mathcal{S}_V(\mathbf{m}_2)$  equals the difference between the augmented variation vectors of  $\mathbf{m}_1$  and  $\mathbf{m}_2$ . With the operator  $\mathcal{S}_V$ , we let the matrix  $\mathbf{B}_k$  in both (30) and (31) be the identity matrix, such that the corresponding Hessian matrices become

$$\nabla_{\|\bullet\|_1^2}^2 \left[ \mathcal{S}_V(\bar{\mathbf{m}}^i) - \mathcal{S}_V(\mathbf{m}_j^i) \right] = 2 \operatorname{s\,gn} \left( \left( \mathcal{S}_V(\bar{\mathbf{m}}^i) - \mathcal{S}_V(\mathbf{m}_j^i) \right) \left( \mathcal{S}_V(\bar{\mathbf{m}}^i) - \mathcal{S}_V(\mathbf{m}_j^i) \right)^T \right); \tag{34}$$

$$\nabla_{\|\bullet\|_2^2}^2 \left[ \mathcal{S}_V(\bar{\mathbf{m}}^i) - \mathcal{S}_V(\mathbf{m}_j^i) \right] = 2\mathbf{I}. \tag{35}$$



**Fig. 1** Reference model used in the synthetic case study, with well positions marked by white ellipses. Well names (in green) starting with the letter “I” stand for injection wells, while those with the letter “P” for production wells



According to Eq. 26, the corresponding PLHs with respect to the reservoir model  $\mathbf{m}_j^i$  are given by

$$\mathbf{M}_{V\ell_1^i}^i(\mathbf{m}_j^i) \equiv (\mathbf{S}_{S_V}^i)^T \text{sgn} \left( (S_V(\bar{\mathbf{m}}^i) - S_V(\mathbf{m}_j^i)) (S_V(\bar{\mathbf{m}}^i) - S_V(\mathbf{m}_j^i))^T \right) \mathbf{S}_{S_V}^i, \text{ (case } \ell_1^i) \quad (36)$$

and

$$\mathbf{M}_{V\ell_2^i}^i(\mathbf{m}_j^i) \equiv (\mathbf{S}_{S_V}^i)^T \mathbf{S}_{S_V}^i, \text{ (case } \ell_2^i), \quad (37)$$

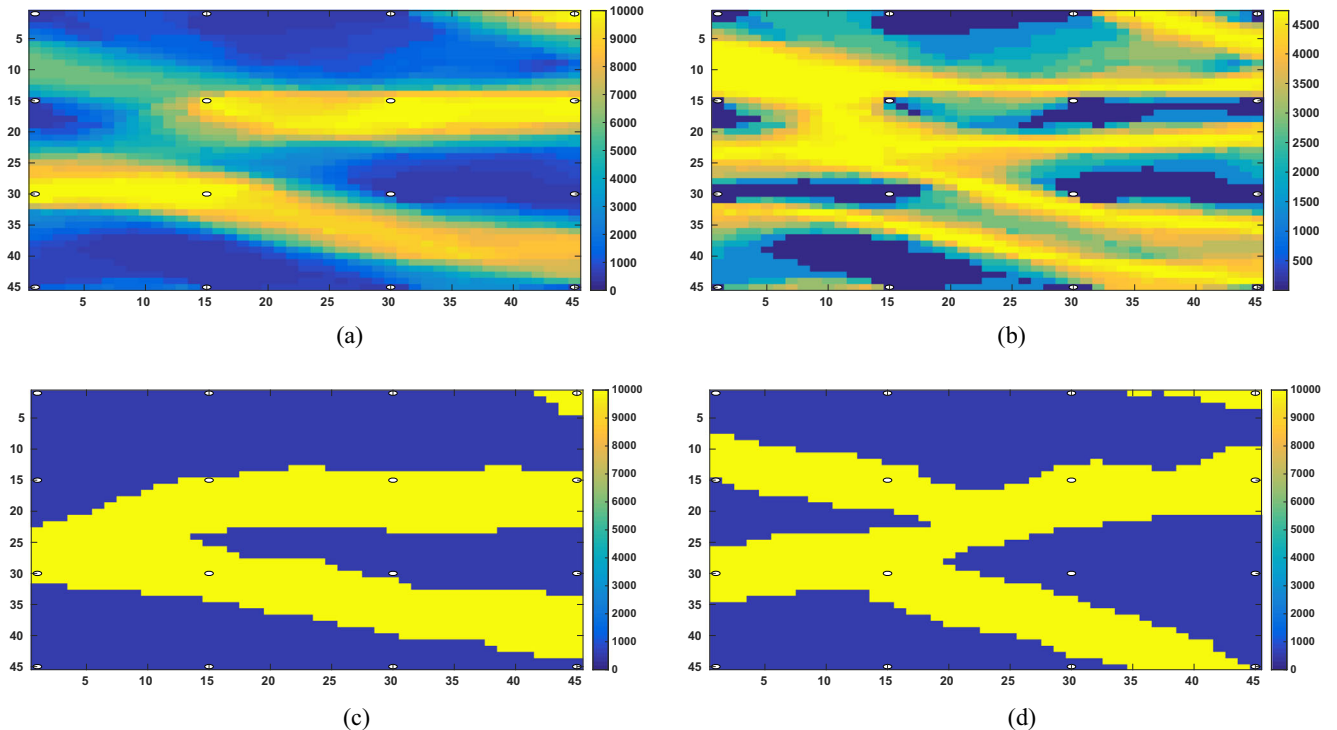
respectively, where the square-root matrix  $\mathbf{S}_{S_V}^i$  is computed according to Eq. 8, with the operator  $\mathcal{O}$  therein replaced by

the operator  $S_V$ , and “V” in the subscript of the PLHs stands for variation. Note that in Eq. 36, the component

$$(S_V(\bar{\mathbf{m}}^i) - S_V(\mathbf{m}_j^i)) (S_V(\bar{\mathbf{m}}^i) - S_V(\mathbf{m}_j^i))^T$$

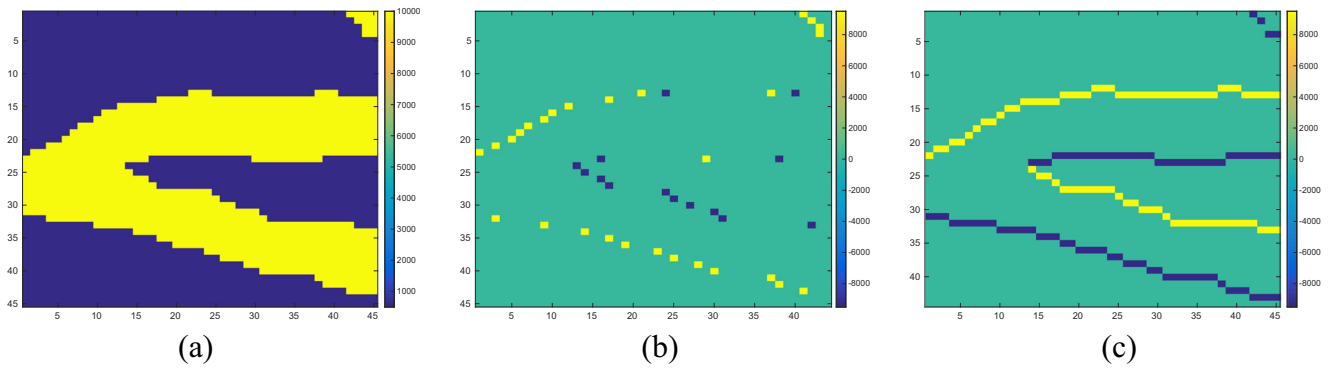
is a rank-1 matrix. As a result, we expect the corresponding PLH  $\mathbf{M}_{V\ell_1^i}^i(\mathbf{m}_j^i)$  may be of much lower rank, in comparison to the PLH  $\mathbf{M}_{V\ell_2^i}^i(\mathbf{m}_j^i)$  in Eq. 37, whose rank would be  $N_e - 1$  given independent ensemble members  $\mathbf{m}_j^i$ .

In S-III, the transform operator first calculates the histogram of a permeability map. For distinction, we denote



**Fig. 2** Mean (top left) and STD (top right) maps of the initial ensemble of permeability (PERM), and two sample reservoir models of the initial ensemble (bottom). **a** Mean map of the initial ensemble of PERM.

**b** STD map of the initial ensemble of PERM. **c** Sample realization 1 of the initial ensemble. **d** Sample realization 2 of the initial ensemble



**Fig. 3** Left: Permeability map of sample realization 1 of the initial ensemble, as also indicated in Fig. 2c; Middle: Spatial variation of the permeability map along the horizontal direction;

Right: Spatial variation of the permeability map along the vertical direction. **a** Sample realization 1 of the initial. **b** Horizontal variations. **c** Vertical variations

this operator by  $\mathcal{S}_H$ . Suppose that the total number of permeability values is  $M_p$  ( $M_p = 45 \times 45 = 2025$  in this case study), the number of bins of the histogram is  $N_{bin}$  (set to 100 in the experiment), and the number of permeability values falling into the  $k$ th bin is  $M_{b_k}$ , such that  $\sum_{k=1}^{N_{bin}} M_{b_k} = M_p$ . For each bin, say, the  $k$ th one, we assign to it a value  $E_k \equiv -M_p p_k \ln(p_k)$ , where  $p_k = M_{b_k}/M_p$  is the empirical probability (or relative frequency) that a permeability value falls into the  $k$ th bin. We note that  $E_k$  can be considered as the component-wise Shannon’s information entropy [30], scaled by the factor  $M_p$ . By definition, we also have  $E_k = M_{b_k} \times (\ln(M_p) - \ln(M_{b_k}))$ , which is the formula used in our computation. For numerical evaluations, we adopt the custom that  $0 \times \ln(0) = 0$ .

Furthermore, the transform operator  $\mathcal{S}_H$  augments the entropies of all histogram bins into a column vector. For two reservoir models, say  $\mathbf{m}_1$  and  $\mathbf{m}_2$ , the difference  $\mathcal{S}_H(\mathbf{m}_1) - \mathcal{S}_H(\mathbf{m}_2)$  is then equal to the difference between the augmented information-entropy vectors of  $\mathbf{m}_1$  and  $\mathbf{m}_2$ . We note that the Hessian and the PLH in S-III bear similarities to those in S-II. We let the matrix  $\mathbf{B}_k$  in both Eqs. 30 and 31 be the identity matrix again, so that the corresponding Hessian matrices in S-III become

$$\nabla_{\|\cdot\|_2}^2 [\mathcal{S}_H(\bar{\mathbf{m}}^i) - \mathcal{S}_H(\mathbf{m}_j^i)] = 2 \, sgn \left( (\mathcal{S}_H(\bar{\mathbf{m}}^i) - \mathcal{S}_H(\mathbf{m}_j^i)) \right) (\mathcal{S}_H(\bar{\mathbf{m}}^i) - \mathcal{S}_H(\mathbf{m}_j^i))^T; \tag{38}$$

$$\nabla_{\|\cdot\|_2}^2 [\mathcal{S}_H(\bar{\mathbf{m}}^i) - \mathcal{S}_H(\mathbf{m}_j^i)] = 2\mathbf{I}, \tag{39}$$

and the corresponding PLHs can be expressed as

$$\mathbf{M}_{H\ell_1^2}^i(\mathbf{m}_j^i) \equiv (\mathbf{S}_{\mathcal{S}_H}^i)^T \, sgn \left( (\mathcal{S}_H(\bar{\mathbf{m}}^i) - \mathcal{S}_H(\mathbf{m}_j^i)) (\mathcal{S}_H(\bar{\mathbf{m}}^i) - \mathcal{S}_H(\mathbf{m}_j^i))^T \right) \mathbf{S}_{\mathcal{S}_H}^i, \text{ (case } \ell_1^2) \tag{40}$$

and

$$\mathbf{M}_{H\ell_2^2}^i(\mathbf{m}_j^i) \equiv (\mathbf{S}_{\mathcal{S}_H}^i)^T \, \mathbf{S}_{\mathcal{S}_H}^i, \text{ (case } \ell_2^2), \tag{41}$$

respectively, where the square-root matrix  $\mathbf{S}_{\mathcal{S}_H}^i$  is again computed according to Eq. 8, and “H” in the subscript of the PLHs stands for histogram. Similar to the situation in S-II, the PLH  $\mathbf{M}_{H\ell_1^2}^i$  is of much lower rank, in comparison to  $\mathbf{M}_{H\ell_2^2}^i$ .

Combing different  $\ell_p^q$  metrics and transform operators, one can establish a set of  $\ell_p^q$ -GIES algorithms according to Eq. 24, with a mixture  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$  of PLHs in the form of

$$\begin{aligned} \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) &= w_1 \mathbf{M}_{I\ell_2^2}^i(\mathbf{m}_j^i) + w_2 \mathbf{M}_{V\ell_2^2}^i(\mathbf{m}_j^i) + \\ &w_3 \mathbf{M}_{V\ell_1^2}^i(\mathbf{m}_j^i) + w_4 \mathbf{M}_{H\ell_2^2}^i(\mathbf{m}_j^i) + w_5 \mathbf{M}_{H\ell_1^2}^i(\mathbf{m}_j^i) \\ &= w_1 \mathbf{I}_{N_e} + w_2 \mathbf{M}_{V\ell_2^2}^i(\mathbf{m}_j^i) + w_3 \mathbf{M}_{V\ell_1^2}^i(\mathbf{m}_j^i) + \\ &w_4 \mathbf{M}_{H\ell_2^2}^i(\mathbf{m}_j^i) + w_5 \mathbf{M}_{H\ell_1^2}^i(\mathbf{m}_j^i). \end{aligned} \tag{42}$$

For ease of reference later, we adopt a 5-bit binary code to represent the  $\ell_p^q$ -GIES algorithms. In each bit, “1” represents the presence of the corresponding PLH, while “0” means the absence of such a PLH. For instance, the code “10000” stands for the original IES with  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) = w_1 \mathbf{I}_{N_e}$ , and “10100” for the  $\ell_p^q$ -GIES algorithm in which  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) = w_1 \mathbf{I}_{N_e} + w_3 \mathbf{M}_{V\ell_1^2}^i(\mathbf{m}_j^i)$ . As a result, following Eq. 42, we end up with  $2^5 - 1 = 31$   $\ell_p^q$ -GIES algorithms in total, after excluding the case with the binary code “00000” (i.e., no regularization). Table 2 lists these considered  $\ell_p^q$ -GIES algorithms, where the information of

**Table 2** Characteristics of the tested  $\ell_p^q$ -GIES algorithms

Rank	Binary code	With the individual regularization term in the cost function?				
		$(\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)^T (\mathbf{C}_m^i)^{-1} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i) \  \mathcal{S}_V (\mathbf{m}_j^{i+1}) - \mathcal{S}_V (\mathbf{m}_j^i) \ _2^2$	$\  \mathcal{S}_V (\mathbf{m}_j^{i+1}) - \mathcal{S}_V (\mathbf{m}_j^i) \ _1^2$	$\  \mathcal{S}_H (\mathbf{m}_j^{i+1}) - \mathcal{S}_H (\mathbf{m}_j^i) \ _2^2$	$\  \mathcal{S}_H (\mathbf{m}_j^{i+1}) - \mathcal{S}_H (\mathbf{m}_j^i) \ _1^2$	
1	11000	Yes	Yes	No	No	No
2	10011	Yes	No	No	Yes	Yes
3	01001	No	Yes	No	No	Yes
4	10010	Yes	No	No	Yes	No
5	01111	No	Yes	Yes	Yes	Yes
6	01011	No	Yes	No	Yes	Yes
7	10001	Yes	No	No	No	Yes
8	11001	Yes	Yes	No	No	Yes
9	01010	No	Yes	No	Yes	No
10	11100	Yes	Yes	Yes	No	No
11	01101	No	Yes	Yes	No	Yes
12	01110	No	Yes	Yes	Yes	No
13	10110	Yes	No	Yes	Yes	No
14	10111	Yes	No	Yes	Yes	Yes
15	10000	Yes	No	No	No	No
16	11010	Yes	Yes	No	Yes	No
17	01100	No	Yes	Yes	No	No
18	10100	Yes	No	Yes	No	No
19	11011	Yes	Yes	No	Yes	Yes
20	10101	Yes	No	Yes	No	Yes
21	01000	No	Yes	No	No	No
22	11101	Yes	Yes	Yes	No	Yes
23	11111	Yes	Yes	Yes	Yes	Yes
24	11110	Yes	Yes	Yes	Yes	No
25	00011	No	No	No	Yes	Yes
26	00010	No	No	No	Yes	No
27	00100	No	No	Yes	No	No
28	00001	No	No	No	No	Yes
29	00101	No	No	Yes	No	Yes
30	00110	No	No	Yes	Yes	No
31	00111	No	No	Yes	Yes	Yes

The algorithms are ranked in terms of mean data mismatch values during the forecast period (see Table 3). Definitions of the matrix  $\mathbf{C}_m^i$  and the transform operators  $\mathcal{S}_V$  and  $\mathcal{S}_H$  are provided in the text, whereas the weights associated with the individual regularization terms are specified in Table 3

presence or absence of individual regularization terms in the cost function is provided.

The PLHs in S-II and S-III are rank-deficient in general, and this deficiency would be particularly serious when using the  $\ell_1^2$  metric, as in  $\mathbf{M}_{V\ell_1^2}$  and  $\mathbf{M}_{H\ell_1^2}$ . Yet, the mixture  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$  of PLHs may still be of full-rank if the PLH  $\mathbf{M}_{I\ell_2^2}^i$  is present, otherwise it will be rank-deficient as well. As such, to compute the inverse of  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$  in Eq. 24, we

first conduct an eigenvalue decomposition on  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$ , whether it is rank-deficient or not. We then keep the leading eigenvalues (and the corresponding eigenvectors) which add up to 99% or more of the sum of all eigenvalues. The inverse of  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$  is a sudo-inverse, computed based on the truncated eigenvalue decomposition, with the corresponding eigenvalues being the inverse of the kept eigenvalues of  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$ .

There could be different ways in specifying the weight coefficients  $w_k, k = 1, 2, \dots, 5$ , in Eq. 42. In the current work, we follow the strategy in [13], and set  $w_k$  as follows:

$$w_1 = \alpha_1; \quad w_2 = \alpha_2 \frac{N_e}{\text{trace}(\mathbf{M}_{V\ell_2^i}^i)}; \quad w_3 = \alpha_3 \frac{N_e}{\text{trace}(\mathbf{M}_{V\ell_1^i}^i)}; \\ w_4 = \alpha_4 \frac{N_e}{\text{trace}(\mathbf{M}_{H\ell_2^i}^i)}; \quad w_5 = \alpha_5 \frac{N_e}{\text{trace}(\mathbf{M}_{H\ell_1^i}^i)}, \quad (43)$$

where  $\alpha_k (k = 1, 2, \dots, 5)$  are some non-negative coefficients satisfying the constraints  $\alpha_k \in [0, 1]; \sum_{k=1}^5 \alpha_k = 1$ . The rationale behind Eq. 43 is to balance the relative weights of individual PLHs, such that the trace of each non-vanishing PLH term is roughly comparable to the trace of  $\mathbf{I}_{N_e}$  (equal to  $N_e$ ). Typically, this strategy can help to avoid the situation that an ill-conditioned PLH term has a dominating weight over other existing terms, and thus improve the numerical stability in algorithm implementation. After  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$  is determined, the value of  $\gamma^i$  in Eq. 24 is chosen following the same rule as in [13].

To mitigate the adverse effects of sampling errors due to the relatively small ensemble size in use, we equip all the  $\ell_p^q$ -GIES algorithms with an automatic and adaptive localization scheme, labeled as RndShfl-GC in [31]. The stopping criteria of the  $\ell_p^q$ -GIES include: (1) the iteration process reaches a maximum of 50 iteration steps; (2) the change of data mismatch values during two consecutive iteration steps is less than 0.01%. An  $\ell_p^q$ -GIES algorithm will stop if either of these two criteria is satisfied.

### 3.2 Experiment results

This sub-section compares the performance of the chosen  $\ell_p^q$ -GIES algorithms, in terms of data mismatch values during the history matching and forecast periods. Given an ensemble  $\mathcal{M}^i = \{\mathbf{m}_j^i\}_{j=1}^{N_e}$  of  $N_e$  reservoir models at the  $i$ th iteration step, and a collection of observations  $\mathbf{d}^o$  (with the associated observation error covariance matrix  $\mathbf{C}_d$ ) within a given period, we compute an ensemble  $\Xi(\mathcal{M}^i)$  of data mismatch values (normalized by the number  $d$  of production data) as follows:

$$\Xi(\mathcal{M}^i) \equiv \left\{ \Phi_j^i | \Phi_j^i = \frac{1}{d} (\mathbf{d}^o - \mathbf{g}(\mathbf{m}_j^i))^T \mathbf{C}_d^{-1} (\mathbf{d}^o - \mathbf{g}(\mathbf{m}_j^i)) \right\}_{j=1}^{N_e}, \quad (44)$$

with  $d = 240$  in this case study.

In the experiment, we update the reservoir models by assimilating the observations within the history matching period, and cross-validate the qualities of the estimated reservoir models by comparing the fitness of the simulated data of the estimated reservoir models to those of the reference model within the forecast period. As a result, in the sequel, we use the (mean) forecast data mismatch as a measure of the quality of an estimated reservoir model.

Table 3 reports the performance of all 31  $\ell_p^q$ -GIES algorithms, in terms of data mismatch values (in the form of mean  $\pm$  STD) during the history matching and forecast periods, and the associated weight coefficients  $\alpha_k$ . For the purpose of demonstration, it would be sufficient to select some set of the  $\alpha_k$  values without fine-tuning to optimize the performance of each individual  $\ell_p^q$ -GIES algorithm. For ease of comparison, the  $\ell_p^q$ -GIES algorithms are listed in an ascending order of their mean data mismatch values within the forecast period, such that a lower rank implies better performance. Before history matching is conducted, the data mismatch values for the initial ensemble are  $24.1399 \pm 9.3804$  and  $31.6418 \pm 15.7455$  within the history matching and forecast periods, respectively.

Some observations can be made based on the results in Table 3. First of all, in this case study, using only one of the five  $\ell_p^q$  metrics, namely,  $(\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)^T (\mathbf{C}_m^i)^{-1} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)$  (equal to  $\|(\mathbf{S}_{\mathcal{L}}^i)^{-T} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)\|_2^2$ ),  $\|\mathcal{S}_V(\mathbf{m}_j^{i+1}) - \mathcal{S}_V(\mathbf{m}_j^i)\|_2^2$ ,  $\|\mathcal{S}_V(\mathbf{m}_j^{i+1}) - \mathcal{S}_V(\mathbf{m}_j^i)\|_1^2$ ,  $\|\mathcal{S}_H(\mathbf{m}_j^{i+1}) - \mathcal{S}_H(\mathbf{m}_j^i)\|_2^2$  or  $\|\mathcal{S}_H(\mathbf{m}_j^{i+1}) - \mathcal{S}_H(\mathbf{m}_j^i)\|_1^2$ , in the regularization term would not result in the best possible performance. Take the original IES (binary code 10000) as an example, it adopts  $\|(\mathbf{S}_{\mathcal{L}}^i)^{-T} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)\|_2^2$  as the regularization term in Eq. 19. As highlighted (in red) in the table, in terms of mean forecast data mismatch, it is ranked the 15th. In contrast, better performance can be obtained by using as the regularization term a certain mixture of the five  $\ell_p^q$  metrics. For instance, a mixture of the metrics  $\|(\mathbf{S}_{\mathcal{L}}^i)^{-T} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)\|_2^2$  and  $\|\mathcal{S}_V(\mathbf{m}_j^{i+1}) - \mathcal{S}_V(\mathbf{m}_j^i)\|_2^2$  (binary code 11000) leads to the best performance in the experiment, with the mean forecast data mismatch being around 20% lower than that of the original IES. The next best performance is achieved by adopting a combination of the metrics  $\|(\mathbf{S}_{\mathcal{L}}^i)^{-T} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)\|_2^2$ ,  $\|\mathcal{S}_H(\mathbf{m}_j^{i+1}) - \mathcal{S}_H(\mathbf{m}_j^i)\|_2^2$  and  $\|\mathcal{S}_H(\mathbf{m}_j^{i+1}) - \mathcal{S}_H(\mathbf{m}_j^i)\|_1^2$  (binary code 10011), or a combination of the metrics  $\|\mathcal{S}_V(\mathbf{m}_j^{i+1}) - \mathcal{S}_V(\mathbf{m}_j^i)\|_2^2$  and  $\|\mathcal{S}_H(\mathbf{m}_j^{i+1}) - \mathcal{S}_H(\mathbf{m}_j^i)\|_1^2$  (binary code 01001), and so on and so forth.

On the other hand, among all five  $\ell_p^q$  metrics, it seems that  $\|(\mathbf{S}_{\mathcal{L}}^i)^{-T} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)\|_2^2$  and  $\|\mathcal{S}_V(\mathbf{m}_j^{i+1}) - \mathcal{S}_V(\mathbf{m}_j^i)\|_2^2$  have the dominant influence on algorithm performance in this particular case study. In fact, among all  $\ell_p^q$ -GIES algorithms that outperform the original IES, either  $\|(\mathbf{S}_{\mathcal{L}}^i)^{-T} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)\|_2^2$  or  $\|\mathcal{S}_V(\mathbf{m}_j^{i+1}) - \mathcal{S}_V(\mathbf{m}_j^i)\|_2^2$  should be absorbed into the regularization term. Otherwise,

**Table 3** Performance of  $\ell_p^q$ -GIES algorithms in terms of data mismatch values during the history matching and forecast periods, which are evaluated with respect to the ensembles of reservoir models at the final iteration steps

Rank	Binary code	History-matching data mismatch	Forecast data mismatch	Weights
		(mean $\pm$ STD)	(mean $\pm$ STD)	( $\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5$ )
1	11000	5.6651 $\pm$ 1.4907	11.2295 $\pm$ 2.1320	(0.8,0.2,0.0,0)
2	10011	5.3117 $\pm$ 1.4697	11.4270 $\pm$ 2.2018	(0.4,0,0,0.3,0.3)
3	01001	4.4136 $\pm$ 1.6987	11.4477 $\pm$ 3.1040	(0,0.5,0,0,0.5)
4	10010	6.7852 $\pm$ 1.7765	11.5952 $\pm$ 2.4197	(0.8,0,0,0.2,0)
5	01111	5.4465 $\pm$ 2.3342	11.7825 $\pm$ 2.6476	(0,0.25,0.25,0.25,0.25)
6	01011	5.3752 $\pm$ 1.4194	11.8857 $\pm$ 2.5042	(0,0.2,0,0.4,0.4)
7	10001	5.7308 $\pm$ 1.6847	12.0399 $\pm$ 1.8364	(0.8,0,0,0.2)
8	11001	6.8318 $\pm$ 1.6571	12.3625 $\pm$ 2.8368	(0.4,0.3,0,0,0.3)
9	01010	5.1996 $\pm$ 1.2851	12.6262 $\pm$ 2.697	(0,0.5,0,0.5,0)
10	11100	5.5461 $\pm$ 1.9041	12.8776 $\pm$ 2.7141	(0.4,0.3,0.3,0,0)
11	01101	5.9634 $\pm$ 3.7488	12.9788 $\pm$ 6.6565	(0,0.4,0.4,0,0.2)
12	01110	5.4560 $\pm$ 1.9779	13.4668 $\pm$ 3.6107	(0,0.4,0.4,0.2,0)
13	10110	6.1372 $\pm$ 2.7614	13.7114 $\pm$ 4.5172	(0.4,0,0.3,0.3,0)
14	10111	6.1220 $\pm$ 2.8016	13.9228 $\pm$ 4.7956	(0.25,0,0.25,0.25,0.25)
<b>15</b>	<b>10000</b>	<b>9.5685 <math>\pm</math> 4.789</b>	<b>14.0500 <math>\pm</math> 4.1426</b>	<b>(1,0,0,0,0)</b>
16	11010	6.8273 $\pm$ 1.9882	14.0611 $\pm$ 2.9402	(0.4,0.3,0,0.3,0)
17	01100	6.1362 $\pm$ 3.6115	14.0987 $\pm$ 8.0402	(0,0.5,0.5,0,0)
18	10100	8.973 $\pm$ 4.3793	14.3571 $\pm$ 4.5162	(0.8,0,0.2,0,0)
19	11011	5.9859 $\pm$ 1.7851	14.3768 $\pm$ 3.1043	(0.25,0.25,0,0.25,0.25)
20	10101	6.7457 $\pm$ 2.0475	14.4018 $\pm$ 4.8584	(0.4,0,0.3,0,0.3)
21	01000	6.9022 $\pm$ 2.8426	14.5511 $\pm$ 6.4825	(0,1,0,0,0)
22	11101	4.6991 $\pm$ 1.8031	14.9612 $\pm$ 3.7313	(0.25,0.25,0.25,0,0.25)
23	11111	5.6104 $\pm$ 2.5165	16.4231 $\pm$ 3.6174	(0.2,0.2,0.2,0.2,0.2)
24	11110	5.2775 $\pm$ 2.2947	16.858 $\pm$ 3.8077	(0.25,0.25,0.25,0.25,0)
25	00011	24.1399 $\pm$ 9.3804	31.6418 $\pm$ 15.7455	(0,0,0.5,0.5)
26	00010	24.1399 $\pm$ 9.3804	31.6418 $\pm$ 15.7455	(0,0,0,1,0)
27	00100	24.1399 $\pm$ 9.3804	31.6419 $\pm$ 15.7456	(0,0,1,0,0)
28	00001	24.1399 $\pm$ 9.3803	31.6419 $\pm$ 15.7457	(0,0,0,0,1)
29	00101	24.1399 $\pm$ 9.3804	31.6419 $\pm$ 15.7457	(0,0,0.5,0.0.5)
30	00110	24.1399 $\pm$ 9.3804	31.6419 $\pm$ 15.7457	(0,0,0.5,0.5,0)
31	00111	24.1399 $\pm$ 9.3803	31.6419 $\pm$ 15.7457	(0,0,0.2,0.4,0.4)

The  $\ell_p^q$ -GIES algorithms are listed in an ascending order of mean values of forecast data mismatch. In particular, performance of the  $\ell_p^q$ -GIES algorithm corresponding to the original IES is highlighted (in red)

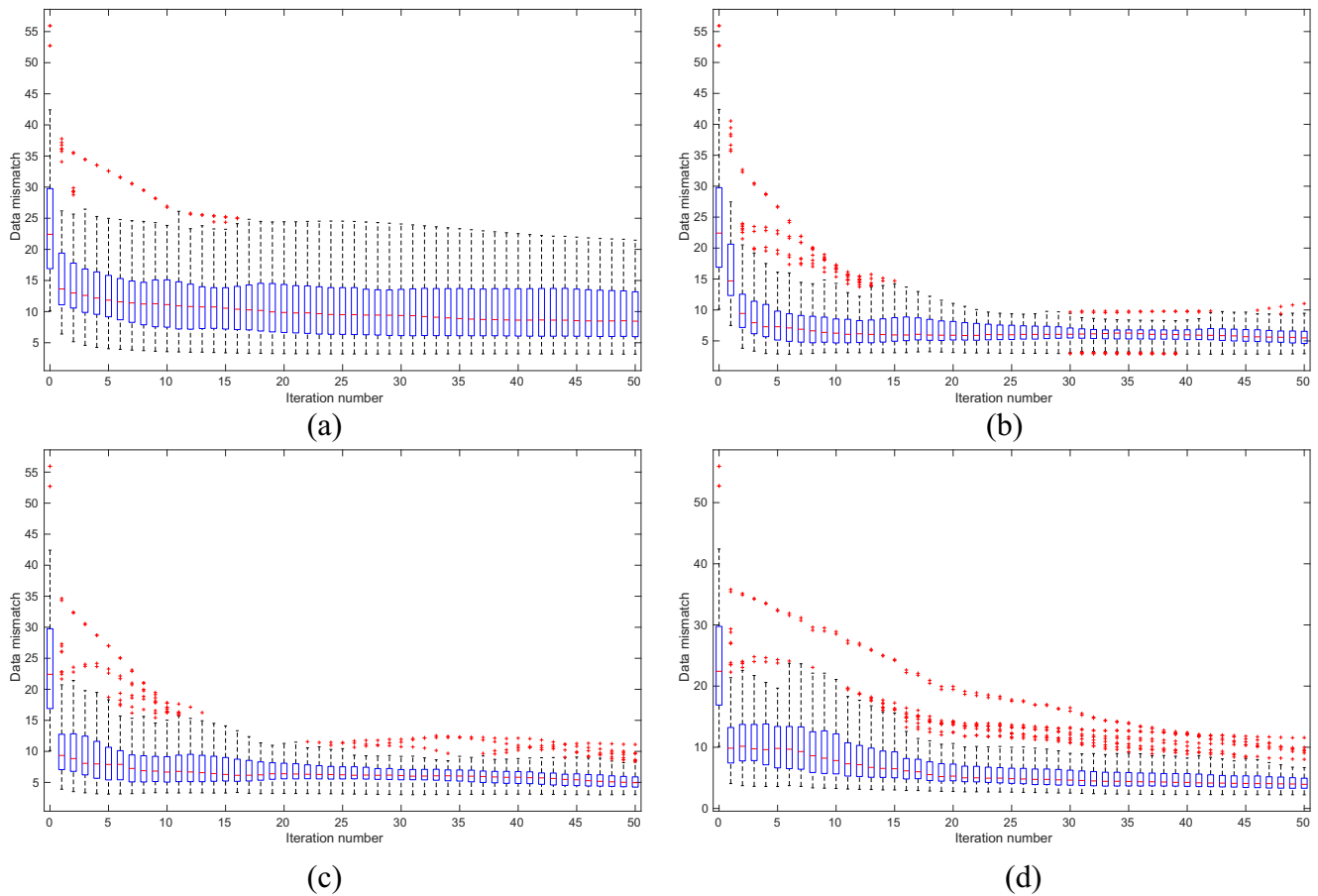
the performance of the  $\ell_p^q$ -GIES algorithms would be degraded, see, for example, the results for those ranked from the 25th to the 31th in the table. As a conjecture, the reason for the under-performance of these  $\ell_p^q$ -GIES algorithms is possibly due to the relatively low ranks in the PLHs  $\mathbf{M}_{V\ell_1^2}^i(\mathbf{m}_j^i)$ ,  $\mathbf{M}_{H\ell_2^2}^i(\mathbf{m}_j^i)$ ,  $\mathbf{M}_{H\ell_1^2}^i(\mathbf{m}_j^i)$ . As discussed previously,  $\mathbf{M}_{V\ell_1^2}^i(\mathbf{m}_j^i)$  and  $\mathbf{M}_{H\ell_1^2}^i(\mathbf{m}_j^i)$  have relatively low ranks due to the presence of the rank-1 matrices in their computations, as can be seen in Eqs. 34 and 38. On the other hand,  $\mathbf{M}_{H\ell_2^2}^i(\mathbf{m}_j^i)$  may also have relatively low ranks due to the possible correlations among the histograms of the initial ensemble of reservoir models, which was generated using the SNESIM with a common training image [28]. With

the low-rank PLHs, the corresponding  $\ell_p^q$ -GIES algorithms are not able to find new reservoir models that significantly reduce the data mismatch values. As a result, one can see that the data mismatch values for reservoir models obtained by these  $\ell_p^q$ -GIES algorithms remain very close to those of the initial ensemble.

For conciseness, in the sequel we present some additional experiment results with respect to some of the selected  $\ell_p^q$ -GIES algorithms, which correspond to binary codes 10000 (original IES, ranked the 15th), 11000 (ranked the first), 10011 (ranked the second) and 01001 (ranked the third), respectively, in Table 3.

Figure 4 reports the box plots of data mismatch within the history matching period at different iteration steps. As indicated in Table 3, in terms of mean data mismatch





**Fig. 4** Box plots of the ensembles of data mismatch values at different iteration steps of history matching, for the  $\ell_p^q$ -GIES algorithms corresponding to binary codes: **a** 10000 (original IES); **b** 11000; **c** 10011 and **d** 01001, respectively

values, the  $\ell_p^q$ -GIES algorithm with the code 01001 results in the lowest mismatch values, and is followed by the  $\ell_p^q$ -GIES algorithms with the codes 10011 and 11000, whereas the original IES (code 10000) leads to the highest data mismatch. Meanwhile, it should be noted that, lower data mismatch within the history matching period does not necessarily imply lower data mismatch within the forecast period. In fact, it is the  $\ell_p^q$ -GIES algorithm with the code 11000 that achieves the lowest mean data mismatch within the forecast period. A possible explanation for this performance-inconsistency is that some of the  $\ell_p^q$ -GIES algorithms may have overfitted the noisy observations within the history matching period, in comparison to others.

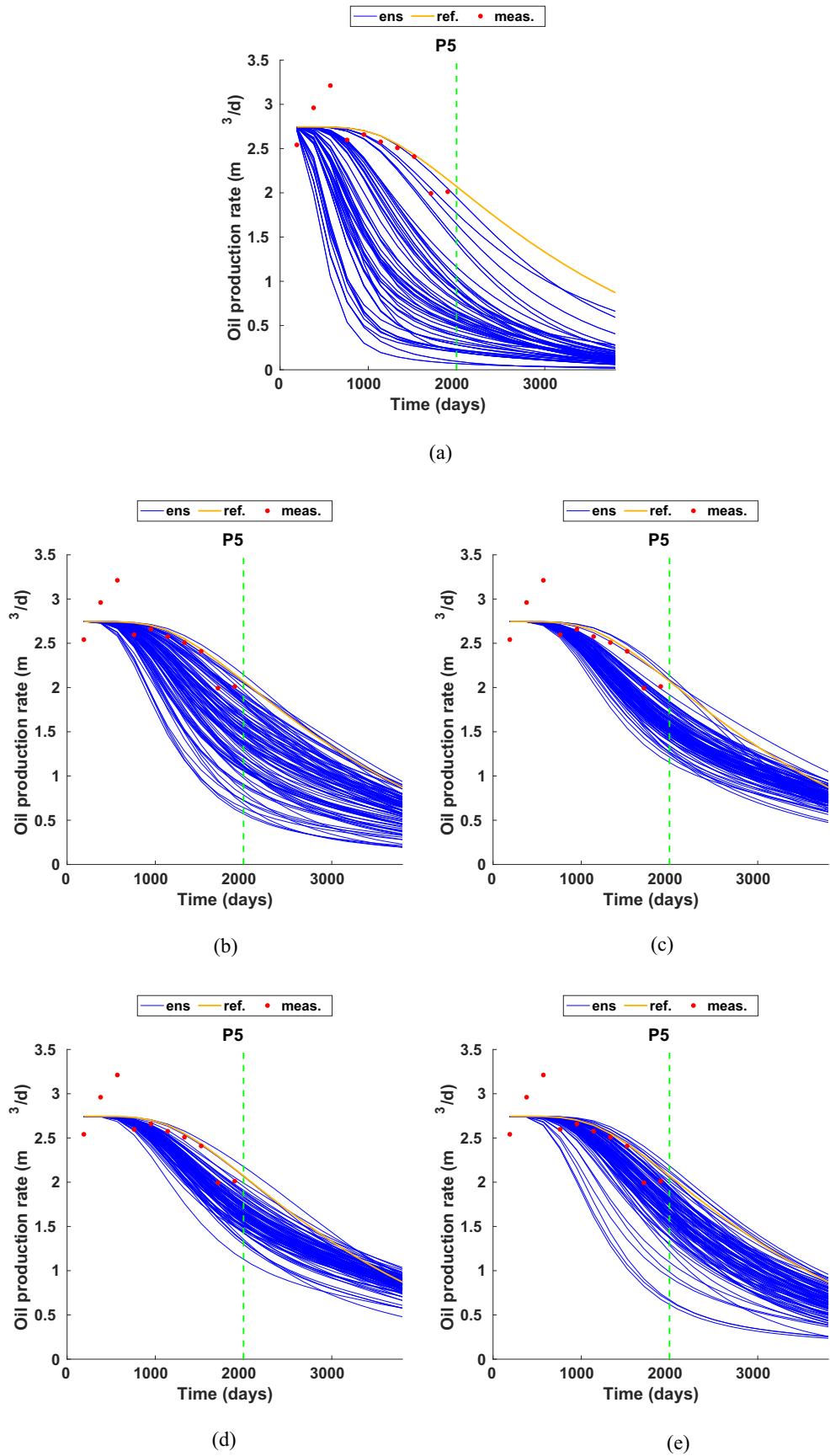
For illustration, Fig. 5 shows the profiles of simulated well oil production rates (WOPR) at P5 within both the history matching and forecast periods, with respect to the initial ensemble and the final ensembles obtained through the selected  $\ell_p^q$ -GIES algorithms, respectively. As can be seen from Fig. 5a, all the simulated WOPR of the initial ensemble (blue curves) under-forecasts the WOPR of the reference model. This is possibly due to the fact that, in the reference model, the rectangular area with the wells

I5, I6, P5 and P6 being the vertices (upper right region of Figs. 1 or 7(a)) contains a relatively large portion of high-permeability values<sup>2</sup>. In contrast, in the initial ensemble, the same rectangular area is filled with more low-permeability values instead (cf Fig. 2a or 7b). After applying the  $\ell_p^q$ -GIES algorithms to update the reservoir models, the simulated data of the final ensembles tend to match the WOPR of the reference model better, although we still see the tendency of under-forecasting the WOPR for most of the updated reservoir models. On the other hand, better data match can be found in other wells, as illustrated in Fig. 6.

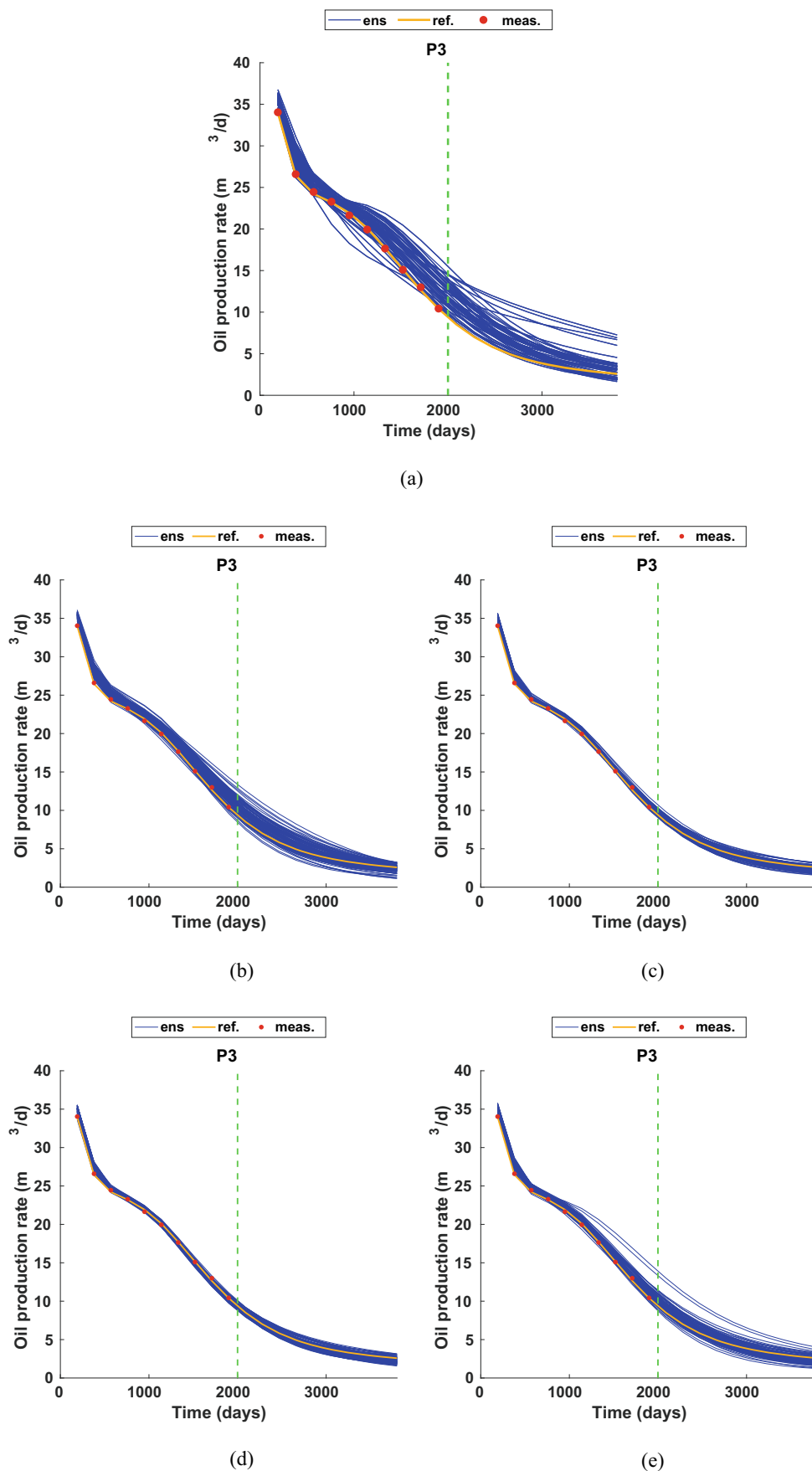
Figure 7 presents the mean permeability maps of the final ensembles obtained by the  $\ell_p^q$ -GIES algorithms. For ease of comparison, we also re-plot the reference model and the mean map of the initial ensemble here. Comparing the mean of the initial ensemble to the reference model, one can see that the initial mean model captures some of the geological structures in the reference model, e.g., in terms

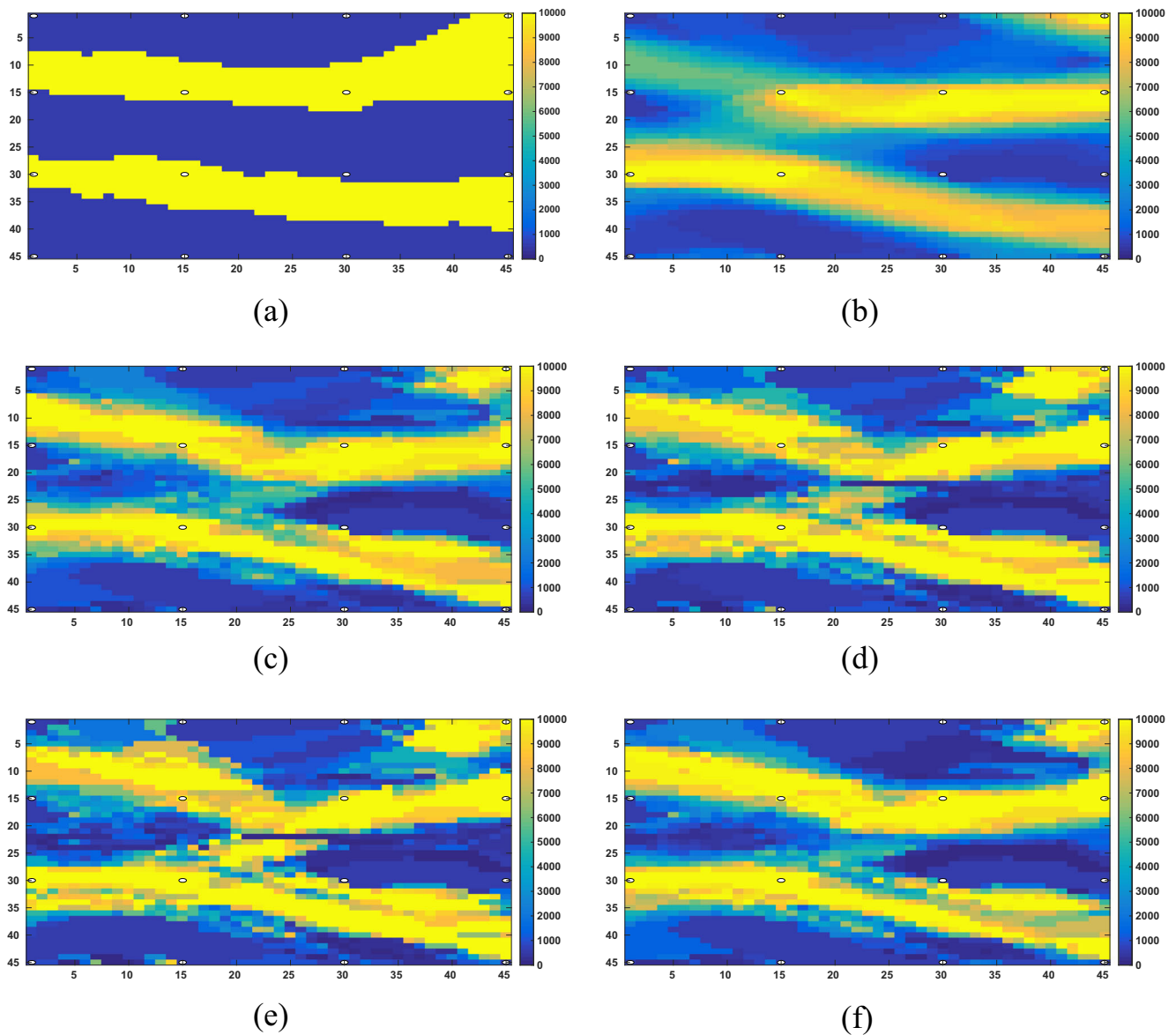
<sup>2</sup>For ease of reference later, we will simply say such an area is surrounded by wells I5, I6, P5 and P6, and adopt the same custom to refer to other areas.

**Fig. 5** Forecasts of WOPR at well P5 within both the history matching and forecast periods. In each sub-figure, the orange curve represents the true WOPR generated by the reference model, without any observation noise; the red dots denote the noisy observations within the history matching period; the blue curves stand for the forecasts from either the initial or the final ensembles of reservoir models (in case of final ensembles, we mark the adopted history matching algorithms by their corresponding binary codes in the captions of sub-figures); and finally, the vertical dashed green lines separate the history matching and forecast periods. **a** Initial ensemble. **b** Binary code 10000 (original IES). **c** Binary code 11000. **d** Binary code 10011. **e** Binary code 01001



**Fig. 6** As in Fig. 5, but for forecasts of WOPR at well P3. **a** Initial ensemble. **b** Binary code 10000 (original IES). **c** Binary code 11000. **d** Binary code 10011. **e** Binary code 01001





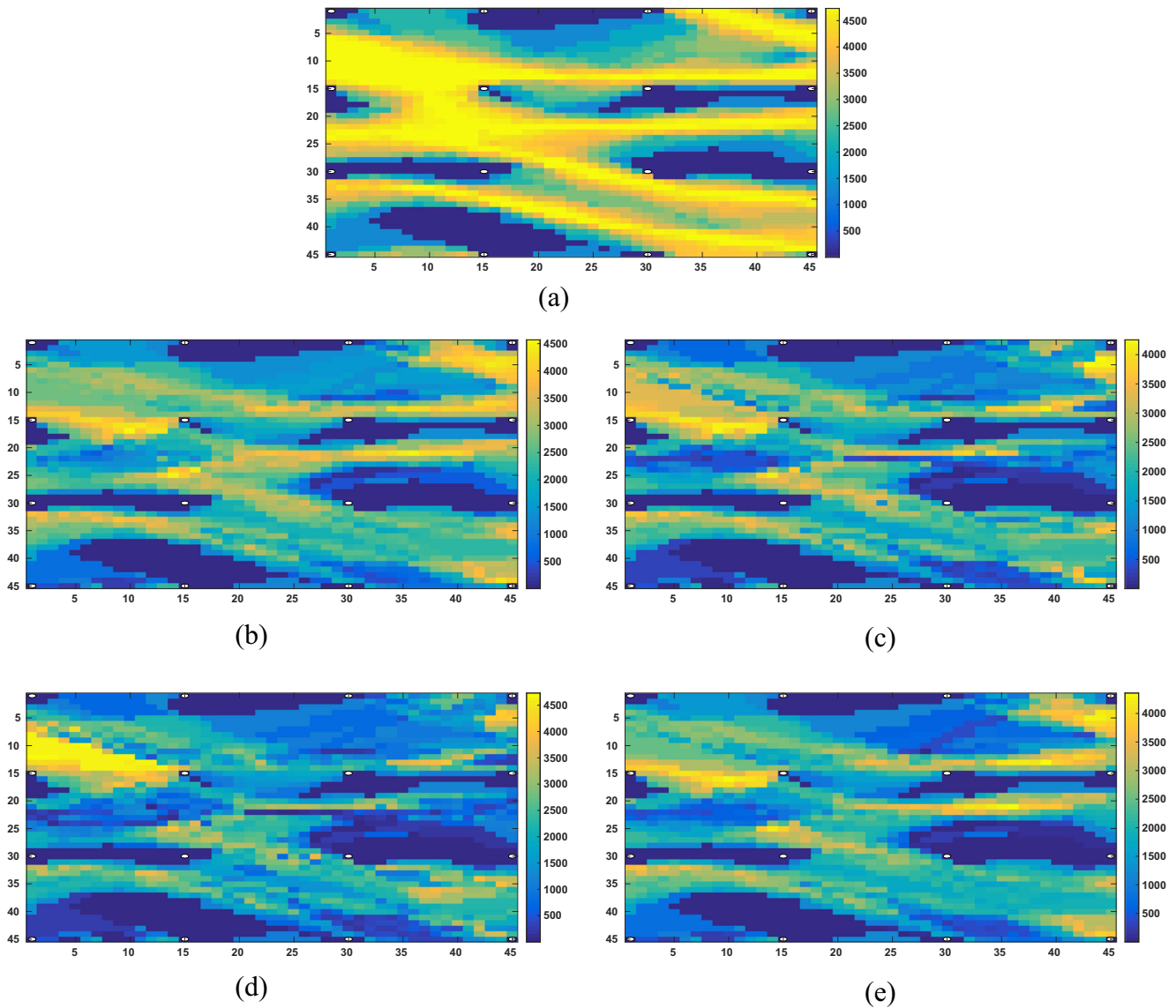
**Fig. 7** Reference model, and mean maps of reservoir models with respect to the initial or final ensemble(s). In case of final ensembles, the captions of corresponding sub-figures indicate the binary codes of

adopted history matching algorithms. **a** Referen model. **b** Initial mean map. **c** Final mean map (code 10000). **d** Final mean map (code 11000). **e** Final mean map (code 10011). **f** Final mean map (code 01001)

of the lower channel closer to the bottom. On the other hand, there are also some noticeable differences, e.g., in the areas surrounded by the wells I1, I2, P1 and P2 (upper left parts of the maps), or the areas surrounded by the wells I5, I6, P5 and P6 (upper right parts of the maps). After history matching, the final mean maps are able to capture the geological structures better in the areas surrounded by the wells I1, I2, P1 and P2, and by the wells I5, I6, P5 and P6, respectively. The connectivities of the channels in the final ensembles of updated reservoir models appear to be a problem. We expect that this issue may be mitigated by adopting a

more sophisticated reparameterization strategy [25–27] in the course of history matching, which is, however, beyond the scope of the current study. For more information, in Figs. 8 and 9 we also report the STD maps with respect to the initial ensemble and final ones obtained by the  $\ell_p^q$ -GIES algorithms, and sample reservoir models before and after history matching, respectively.

Finally, Fig. 10 plots the histograms with respect to the reference model, and the mean models of the initial and final ensembles. While the histogram of the reference model takes two separate points as its support, those of



**Fig. 8** As in Fig. 7, but for STD maps instead. **a** Initial STD map. **b** Final STD map (code 10000). **c** Final STD map (code 11000). **d** Final STD map (code 10011). **e** Final STD map (code 01001)

the mean models of the initial and final ensembles exhibit (approximately) bi-modal distributions. Roughly speaking, the two main modes of the distributions appear to coincide with the true permeability values of the channel (10000 md) and the background (500 md), although their magnitudes (defined as the numbers of permeability values falling into the bins at the modes) seem to vary. For instance, for the dominant mode (10000 md), the  $\ell_p^q$ -GIES algorithms with the codes 11000 and 10011 lead to higher magnitudes ( $> 300$ ) than those with the codes 10000 and 01001. On the other hand, regarding the magnitudes of the secondary modes (500 md), the  $\ell_p^q$ -GIES algorithm with the code 01001 results in the highest value ( $> 100$ ), whereas the

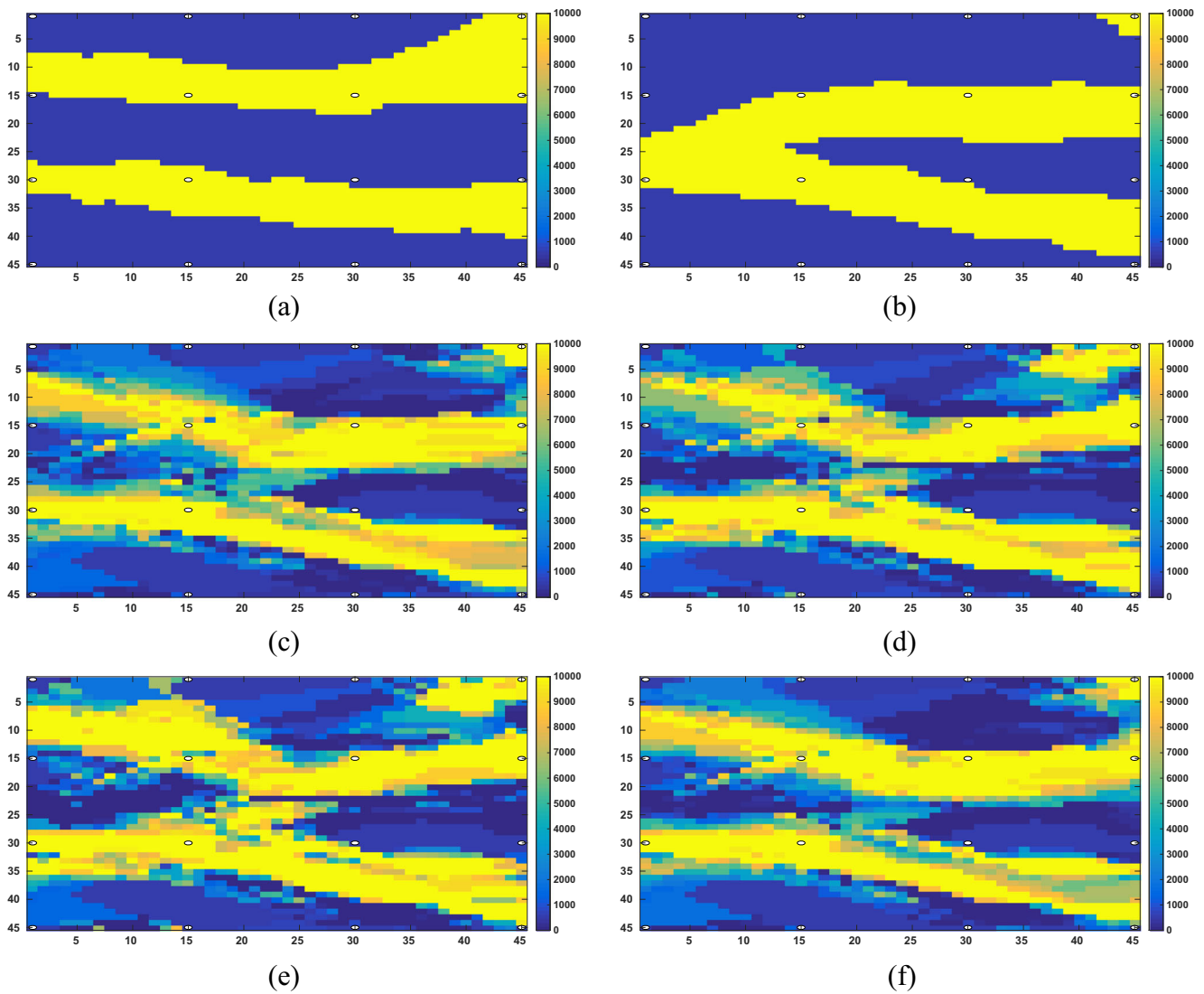
magnitudes of the other  $\ell_p^q$ -GIES algorithms remain close.

### 4 Discussion and conclusion

This work investigates the feasibility to derive novel ensemble data assimilation algorithms in a principled and systematic way, and applies some newly derived algorithms to two history matching problems for performance comparison.

We start from establishing an ensemble-based model update formula, which can be considered as an approximate solution to a generalized minimum-average-cost (GMAC) problem. In the GMAC problem, one aims to find an





**Fig. 9** An individual reservoir model (sample realization 1, cf. Figure 2) before and after history matching. For ease of comparison, the reference model and the binary codes of adopted history matching algorithms are also presented, similar to Fig. 7. **a** Referen model.

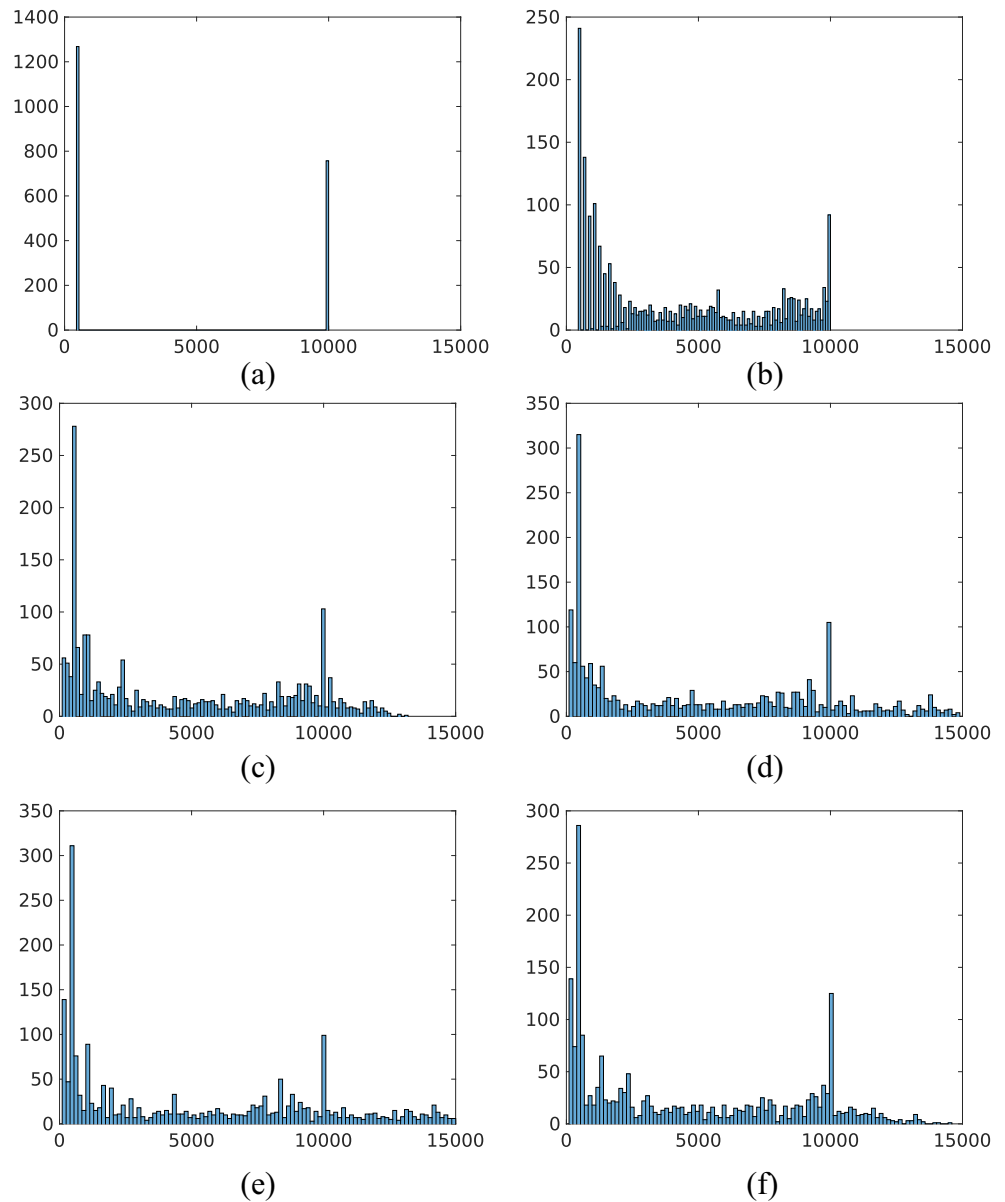
**b** Initial map (sample realization 1). **c** Final mean map (sample realization 1, code 10000). **d** Final mean map (sample realization 1, code 11000). **e** Final mean map (sample realization 1, code 10011). **f** Final mean map (sample realization 1, code 01001)

ensemble of inputs (e.g., model parameters) to a given forward simulator, in such a way that the corresponding outputs of the simulator can match the observed data to a good extent. As such, solving the GMAC problem corresponds to finding an ensemble of solutions to the underlying inverse problem. Similar to the transition from variational data assimilation to ensemble-based data assimilation, the solution to the GMAC problem can be deemed as an “ensemblized” version of a certain deterministic approach to solving the inverse problem. However, in the literature, various deterministic inversion algorithms have been developed by minimizing different types of (regularized) cost functions. In contrast, for ensemble-based data assimilation algorithms in practical

use, such a variety (in terms of the underlying cost functions) does not seem to exist, to the best of our knowledge. The current work narrows this noticed gap between the varieties of the deterministic and ensemble-based inversion algorithms, by showing that one can derive novel ensemble data assimilation algorithms from a class of generalized cost functions, which in general go beyond the form of nonlinear-least-squares in the conventional ensemble-based data assimilation algorithms.

For demonstration, we investigate a sub-class of newly derived ensemble data assimilation algorithms, called the  $\ell_p^q$ -GIES algorithms. The underlying cost functions of these algorithms share the same data mismatch term, but differ in their regularization terms, which consist of

**Fig. 10** Similar to Fig. 7, but for histograms of the reference model, and mean reservoir models with respect to the initial and final ensembles. **a** Reference model. **b** Initial mean map. **c** Final mean map (code 10000). **d** Final mean map (code 11000). **e** Final mean map (code 10011). **f** Final mean map (code 01001)



mixtures of various  $\ell_p^q$  metrics of different transformed model variables. It can be shown that the original iterative ensemble smoother (IES) corresponds to a special case of the  $\ell_p^q$ -GIES algorithms. In applications of the  $\ell_p^q$ -GIES algorithms to two history matching problems, we show that many tested  $\ell_p^q$ -GIES algorithms outperform the original IES in these case studies. In a bigger picture, given the richness of the ensemble-based data assimilation algorithms that can be derived from the general model update formula, it appears sensible to believe that one may find certain algorithms that perform better than the original IES in

various inverse problems. On the other hand, though, how to find the optimal designs of such ensemble-based algorithms would remain to be an open research question.

### Appendix A: Iterative ensemble smoothers derived from a class of generalized cost functions

In the sequel, we proceed to develop an approximate solution to the generalized MAC problem in Eqs. 9 and 10,

in a way similar to that in [13]. To this end, we need to assume certain regularity conditions, namely, the operators  $\mathcal{D}$ ,  $\mathcal{R}$ ,  $\mathcal{T}$ ,  $\mathcal{S}$  and  $\mathbf{g}$  are (locally) differentiable up to relevant orders in our derivations below.

We start from considering the first order Taylor approximation

$$\mathbf{f}(\mathbf{x}_0 + \delta\mathbf{x}) \approx \mathbf{f}(\mathbf{x}_0) + [\nabla_f \mathbf{f}(\mathbf{x}_0)]^T \delta\mathbf{x}, \tag{45}$$

$$\nabla_f \mathbf{f}(\mathbf{x}_0) \equiv \left. \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_0}, \tag{46}$$

where  $\mathbf{f}$  is a vector-valued function,  $\mathbf{x}_0$  is an input vector associated with a (relatively small) perturbation vector  $\delta\mathbf{x}$ , and  $\nabla_f \mathbf{f}(\mathbf{x}_0)$ , as defined in Eq. 46, represents the gradient with respect to  $\mathbf{f}$  evaluated at the point  $\mathbf{x}_0$ . Note that throughout this work, we adopt the following convention: given the  $m_x$ -dimensional vector  $\mathbf{x}_0$  and the function  $\mathbf{f} : \mathbb{R}^{m_x} \rightarrow \mathbb{R}^{m_f}$ , the gradient  $\nabla_f \mathbf{f}(\mathbf{x}_0)$  is a matrix in the dimension of  $m_x \times m_f$ .

The following formulas of matrix calculus [32]

$$\frac{\partial \mathbf{f}(\mathbf{u}(\mathbf{x}))}{\partial \mathbf{x}} = \frac{\partial \mathbf{u}(\mathbf{x})}{\partial \mathbf{x}} \frac{\partial \mathbf{f}(\mathbf{u})}{\partial \mathbf{u}} = \nabla_u \mathbf{f}(\mathbf{u}) \nabla_u \mathbf{u}(\mathbf{x}), \tag{47}$$

$$\frac{\partial \mathbf{M}\mathbf{u}(\mathbf{x})}{\partial \mathbf{x}} = \frac{\partial \mathbf{u}(\mathbf{x})}{\partial \mathbf{x}} \mathbf{M}^T = \nabla_u \mathbf{u}(\mathbf{x}) \mathbf{M}^T. \tag{48}$$

will get involved in our derivation later. In Eq. 47,  $\mathbf{f}(\mathbf{u}(\mathbf{x}))$  represents the composition of two vector functions  $\mathbf{f}$  and  $\mathbf{u}$ , with the dummy input vector  $\mathbf{x}$ . Equation 48 may be considered as a special case of Eq. 47, where  $\mathbf{f}(\mathbf{u}(\mathbf{x})) = \mathbf{M}\mathbf{u}(\mathbf{x})$ , with  $\mathbf{M}$  being a constant matrix, such that  $\partial \mathbf{M}\mathbf{u} / \partial \mathbf{u} = \mathbf{M}^T$ .

In addition, the follow matrix identity

$$\mathbf{C}_m \mathbf{H}^T (\mathbf{H} \mathbf{C}_m \mathbf{H}^T + \mathbf{C}_d)^{-1} = (\mathbf{C}_m^{-1} + \mathbf{H}^T \mathbf{C}_d^{-1} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{C}_d^{-1}, \tag{49}$$

---


$$\begin{aligned} \partial \mathcal{A} / \partial \mathbf{m}_j^{i+1} &= -\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \nabla_{\mathcal{D}} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i) - [\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i)]^T (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i) \right] \\ &\approx -\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \left\{ \nabla_{\mathcal{D}} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i) \right] - \left( \nabla_{\nabla_{\mathcal{D}}} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i) \right] \right)^T \left[ \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \right]^T (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i) \right\} \\ &\equiv -\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \left\{ \nabla_{\mathcal{D}} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i) \right] - \left( \nabla_{\mathcal{D}}^2 \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i) \right] \right)^T \left[ \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \right]^T (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i) \right\}. \end{aligned} \tag{54}$$


---

In Eq. 54, we use Eqs. 47 and 48 to derive the first line. In the second line, we then apply the first Taylor approximation (45), around the point  $\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)$ , to the function  $\nabla_{\mathcal{D}}$  in the first line. As a result, we come out with a second-order gradient (i.e., Hessian)  $\nabla_{\nabla_{\mathcal{D}}}$ , which is denoted as  $\nabla_{\mathcal{D}}^2$  in the third line for short.

will also be found useful for our derivation later. The left-hand-side (LHS) of Eq. 49 would correspond to the Kalman gain matrix in the Kalman filter, if one treats  $\mathbf{C}_m$  as the prior model error covariance matrix,  $\mathbf{H}$  as a linear observation operator that maps a model vector onto the observation space, and  $\mathbf{C}_d$  as the observation error covariance matrix. The right-hand-side (RHS) of Eq. 49 is another way to represent the Kalman gain matrix, and is often used to formulate the information filter [33].

To solve the minimization problem in Eqs. 9–10, we aim to set

$$\partial \mathcal{C}_j^{i+1} / \partial \mathbf{m}_j^{i+1} = 0. \tag{50}$$

To this end, the follow linearization [13] strategy, through the first Taylor approximation (45), is adopted.

$$\mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^{i+1}) \equiv \mathcal{T}(\mathbf{g}(\mathbf{m}_j^{i+1})) \approx \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i) + [\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i)]^T (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i), \tag{51}$$

$$\mathcal{S}(\mathbf{m}_j^{i+1}) \approx \mathcal{S}(\mathbf{m}_c^i) + [\nabla_{\mathcal{S}}(\mathbf{m}_c^i)]^T (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i), \tag{52}$$

where  $\mathbf{m}_c^i$  is a “common” point with respect to the (background) ensemble  $\mathcal{M}^i$ . The motivation to use this “common” point is to avoid evaluating the gradients with respect to the operators  $\mathcal{T} \circ \mathbf{g}$  and  $\mathcal{S}$  at multiple points. There could be various choices for  $\mathbf{m}_c^i$ , e.g., by setting  $\mathbf{m}_c^i$  as the ensemble mean of  $\mathcal{M}^i$ , or as one ensemble member closest to the ensemble mean [13]. In the current work, by default we will let  $\mathbf{m}_c^i$  be the ensemble mean of  $\mathcal{M}^i$ , unless otherwise stated. We note that the choice of  $\mathbf{m}_c^i$  does not affect the deviations below.

Inserting (51) into the data mismatch term  $\mathcal{D}$  of Eq. 10, then we have

$$\mathcal{A} \equiv \mathcal{D} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T}(\mathbf{g}(\mathbf{m}_j^{i+1})) \right] \approx \tag{53}$$

$$\mathcal{D} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i) - [\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i)]^T (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i) \right].$$

Accordingly, we have

Similarly, by inserting (52) into the regularization term  $\mathcal{R}$  of Eq. 10, we have

$$\mathcal{B} \equiv \mathcal{R} \left[ \mathcal{S}(\mathbf{m}_j^{i+1}) - \mathcal{S}(\mathbf{m}_j^i) \right] \approx \tag{55}$$

$$\mathcal{R} \left[ \mathcal{S}(\mathbf{m}_c^i) - \mathcal{S}(\mathbf{m}_j^i) + [\nabla_{\mathcal{S}}(\mathbf{m}_c^i)]^T (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i) \right],$$

and the corresponding derivative is

$$\partial \mathcal{B} / \partial \mathbf{m}_j^{i+1} \approx \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \left\{ \nabla_{\mathcal{R}} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)] + (\nabla_{\mathcal{R}}^2 [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)])^T [\nabla_{\mathcal{S}}(\mathbf{m}_c^i)]^T (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i) \right\}. \tag{56}$$

Combing (10), (50), (54) and (56) and with some linear algebra, one has

$$\begin{aligned} & \left\{ \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) (\nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)])^T [\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i)]^T \right. \\ & \quad \left. + \gamma^i \nabla_{\mathcal{S}}(\mathbf{m}_c^i) (\nabla_{\mathcal{R}}^2 [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)])^T [\nabla_{\mathcal{S}}(\mathbf{m}_c^i)]^T \right\} (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i) \\ = & \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)] - \gamma^i \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \nabla_{\mathcal{R}} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)]. \end{aligned} \tag{57}$$

As in the ensemble-based methods, the model variables  $\mathbf{m}_j^{i+1}$  are updated from  $\mathbf{m}_j^i$ , rather than from  $\mathbf{m}_c^i$ , we rearrange (57) as follows:

$$\begin{aligned} & \left\{ \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) (\nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)])^T [\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i)]^T \right. \\ & \quad \left. + \gamma^i \nabla_{\mathcal{S}}(\mathbf{m}_c^i) (\nabla_{\mathcal{R}}^2 [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)])^T [\nabla_{\mathcal{S}}(\mathbf{m}_c^i)]^T \right\} (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i) \\ = & \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)] - \gamma^i \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \nabla_{\mathcal{R}} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)] \\ & + \left\{ \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) (\nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)])^T [\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i)]^T \right. \\ & \quad \left. + \gamma^i \nabla_{\mathcal{S}}(\mathbf{m}_c^i) (\nabla_{\mathcal{R}}^2 [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)])^T [\nabla_{\mathcal{S}}(\mathbf{m}_c^i)]^T \right\} (\mathbf{m}_c^i - \mathbf{m}_j^i). \end{aligned} \tag{58}$$

After regrouping different terms, the RHS of Eq. 58 can be rewritten as  $\mathcal{E} + \mathcal{F}$ , with

$$\begin{aligned} \mathcal{E} & \equiv \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \left\{ \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)] + (\nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)])^T [\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i)]^T (\mathbf{m}_c^i - \mathbf{m}_j^i) \right\} \\ & \approx \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)] + [\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i)]^T (\mathbf{m}_c^i - \mathbf{m}_j^i) \\ & \approx \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)]. \end{aligned} \tag{59}$$

In Eq. 59, the second line is obtained by applying the Taylor approximation (45) (in a reverse order) to the function  $\nabla_{\mathcal{D}}$ , around the point  $\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)$ .

Likewise, the third line is derived by applying (45) again, now to the function  $\mathcal{T} \circ \mathbf{g}$ , around the point  $\mathbf{m}_c^i$ .

Likewise, we have

$$\begin{aligned} \mathcal{F} & \equiv -\gamma^i \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \left\{ \nabla_{\mathcal{R}} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)] - (\nabla_{\mathcal{R}}^2 [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)])^T [\nabla_{\mathcal{S}}(\mathbf{m}_c^i)]^T (\mathbf{m}_c^i - \mathbf{m}_j^i) \right\} \\ & \approx -\gamma^i \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \nabla_{\mathcal{R}} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)] - [\nabla_{\mathcal{S}}(\mathbf{m}_c^i)]^T (\mathbf{m}_c^i - \mathbf{m}_j^i) \\ & \approx -\gamma^i \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \nabla_{\mathcal{R}} [S(\mathbf{m}_j^i) - S(\mathbf{m}_j^i)] \\ & = \mathbf{0}. \end{aligned} \tag{60}$$

To derive the final result in Eq. 60, we make the following assumption:  $\nabla_{\mathcal{R}}(\mathbf{0}) = \mathbf{0}$ . The rationale behind this assumption is that, in a conventional setting of regularized inverse-problem theory [6], the regularization term would typically achieve the minimum value (and actually vanish) at the zero value.

Combining (58) – (60), one obtains

$$\begin{aligned} & \left\{ \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) (\nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)])^T [\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i)]^T \right. \\ & \quad \left. + \gamma^i \nabla_{\mathcal{S}}(\mathbf{m}_c^i) (\nabla_{\mathcal{R}}^2 [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)])^T [\nabla_{\mathcal{S}}(\mathbf{m}_c^i)]^T \right\} (\mathbf{m}_j^{i+1} - \mathbf{m}_c^i) \\ = & \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)]. \end{aligned} \tag{61}$$

As the Hessian  $\nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)]$  and  $\nabla_{\mathcal{R}}^2 [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)]$  are symmetric, we discard the transpose operator  $T$  hereafter. To proceed further, we need to apply (49)–(61). To this end, we do the following assignments:

$$(\mathbf{C}_{m,j}^i)^{-1} = \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \nabla_{\mathcal{R}}^2 [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)] [\nabla_{\mathcal{S}}(\mathbf{m}_c^i)]^T; \tag{62}$$

$$(\mathbf{C}_{d,j}^i)^{-1} = \nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)]; \tag{63}$$

$$\mathbf{H}^i = [\nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i)]^T. \tag{64}$$

Then using Eq. 49 and with some algebra, one has the following update formula:

$$\begin{aligned} \mathbf{m}_j^{i+1} &= \mathbf{m}_j^i + \mathbf{K}_j^i \mathbf{C}_{d,j}^i \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)] \\ &= \mathbf{m}_j^i + \mathbf{K}_j^i \left\{ \nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)] \right\}^{-1} \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)], \quad (65) \\ \mathbf{K}_j^i &\equiv \mathbf{C}_{m,j}^i (\mathbf{H}^i)^T \left( \mathbf{H}^i \mathbf{C}_{m,j}^i (\mathbf{H}^i)^T + \gamma^i \mathbf{C}_{d,j}^i \right)^{-1}. \quad (66) \end{aligned}$$

Before proceeding further, we have a few remarks. First, the assimilation algorithm presented in Eqs. 62–66 provides an approximation solution to the GMAC problem, but without using ensemble approximations to the first- and second-order gradients therein yet. If one already has access to all required gradients, then they can be used in the assimilation algorithm, and this may lead to improved performance of history matching. On the other hand, if some of the gradients are not available or impractical to compute and/or store, then one may employ ensemble approximations to obtain certain (sometimes partially) derivative-free algorithms, as will be discussed later.

Second, in the original IES algorithm (3)–(7), the Kalman-gain-like matrix,  $\mathbf{K}^i$  in Eq. 4, is common to all ensemble members. In contrast, in the case with a generalized cost function, e.g., that in Eq. 10, the corresponding Kalman-gain-like matrix,  $\mathbf{K}_j^i$  in Eq. (66),

depends on both the individual ensemble member  $\mathbf{m}_j^i$  and even the (potentially) perturbed data  $\mathbf{d}_j$  in general, as one can see from Eqs. 62–63. While the diversity of  $\mathbf{K}_j^i$  may be of theoretical interest, in practice one of the implications is the increased memory consumption for storing these Kalman-gain-like matrices. This problem can be avoided if one lets both the distance metric  $\mathcal{D}$  and the regularization operator  $\mathcal{R}$  be some quadratic functions, as in the problem formulation that leads to the original IES, cf. Equations 2 or Eqs. 11 and 12; or partially mitigated if one adopts ensemble approximations to compute  $\mathbf{K}_j^i$ , as will be shown below.

To obtain ensemble-based approximation, we start from computing the product  $\mathbf{C}_{m,j}^i (\mathbf{H}^i)^T$  involved in the computation of  $\mathbf{K}_j^i$ . Let

$$\begin{aligned} \mathbf{T}_1 &\equiv \mathbf{C}_{m,j}^i (\mathbf{H}^i)^T \\ &= \left\{ \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \nabla_{\mathcal{R}}^2 [\mathcal{S}(\mathbf{m}_c^i) - \mathcal{S}(\mathbf{m}_j^i)] \left[ \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \right]^T \right\}^{-1} \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i) \\ &= \mathbf{S}_{\mathcal{I}}^i \left\{ (\mathbf{S}_{\mathcal{I}}^i)^T \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \nabla_{\mathcal{R}}^2 [\mathcal{S}(\mathbf{m}_c^i) - \mathcal{S}(\mathbf{m}_j^i)] \left[ \nabla_{\mathcal{S}}(\mathbf{m}_c^i) \right]^T \mathbf{S}_{\mathcal{I}}^i \right\}^{-1} \quad (67) \\ &\quad (\mathbf{S}_{\mathcal{I}}^i)^T \nabla_{\mathcal{T} \circ \mathbf{g}}(\mathbf{m}_c^i), \end{aligned}$$

where  $\mathbf{S}_{\mathcal{I}}$  is an (ensemble-induced) square-root matrix, as defined in Eqs. 6 and 8. Now, for a generic function  $\mathbf{f}$  with suitable regularity conditions, we have

$$\begin{aligned} (\mathbf{S}_{\mathcal{I}}^i)^T \nabla_{\mathbf{f}}(\mathbf{m}_c^i) &= \frac{1}{\sqrt{N_e - 1}} \left[ (\mathbf{m}_1^i - \bar{\mathbf{m}}^i)^T \nabla_{\mathbf{f}}(\mathbf{m}_c^i), (\mathbf{m}_2^i - \bar{\mathbf{m}}^i)^T \nabla_{\mathbf{f}}(\mathbf{m}_c^i), \dots, (\mathbf{m}_{N_e}^i - \bar{\mathbf{m}}^i)^T \nabla_{\mathbf{f}}(\mathbf{m}_c^i) \right] \\ &\approx \frac{1}{\sqrt{N_e - 1}} \left[ \mathbf{f}(\mathbf{m}_1^i) - \mathbf{f}(\mathbf{m}_c^i), \mathbf{f}(\mathbf{m}_2^i) - \mathbf{f}(\mathbf{m}_c^i), \dots, \mathbf{f}(\mathbf{m}_{N_e}^i) - \mathbf{f}(\mathbf{m}_c^i) \right]^T \\ &\equiv (\mathbf{S}_{\mathbf{f}}^i)^T. \quad (68) \end{aligned}$$

To drive the results in Eq. 68, we have used the fact that  $\bar{\mathbf{m}}^i = \mathbf{m}_c^i$ , and applied the first order Taylor approximation (45), in the reverse order, to derive the result in the second line.

With Eq. 68, we then have

$$\begin{aligned} \mathbf{T}_1 &\approx \mathbf{S}_{\mathcal{I}}^i \left\{ (\mathbf{S}_{\mathcal{S}}^i)^T \nabla_{\mathcal{R}}^2 [\mathcal{S}(\mathbf{m}_c^i) - \mathcal{S}(\mathbf{m}_j^i)] \mathbf{S}_{\mathcal{S}}^i \right\}^{-1} (\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i)^T \\ &= \mathbf{S}_{\mathcal{I}}^i \left[ \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) \right]^{-1} (\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i)^T. \quad (69) \end{aligned}$$

$$\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) \equiv (\mathbf{S}_{\mathcal{S}}^i)^T \nabla_{\mathcal{R}}^2 [\mathcal{S}(\mathbf{m}_c^i) - \mathcal{S}(\mathbf{m}_j^i)] \mathbf{S}_{\mathcal{S}}^i. \quad (70)$$

If one does not have the analytic form of the Hessian  $\nabla_{\mathcal{R}}^2$  of the regularization operator  $\mathcal{R}$ , then an ensemble-based approximation strategy can be further deployed to compute  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$ , which will be presented separately later.

Now we consider the computation of the product  $(\mathbf{H}^i) \mathbf{C}_{m,j}^i (\mathbf{H}^i)^T$ . Following the previous deduction, we have

$$\begin{aligned} \mathbf{T}_2 &\equiv \mathbf{H}^i \mathbf{C}_{m,j}^i (\mathbf{H}^i)^T \\ &\approx \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \left[ \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) \right]^{-1} (\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i)^T. \quad (71) \end{aligned}$$

On the other hand, we have

$$\begin{aligned} \mathbf{C}_{d,j}^i &= \left\{ \nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)] \right\}^{-1} \\ &= \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \left\{ (\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i)^T \nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)] \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right\}^{-1} (\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i)^T \\ &= \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \left[ \mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) \right]^{-1} (\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i)^T \quad (72) \end{aligned}$$

$$\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) \equiv (\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i)^T \nabla_{\mathcal{D}}^2 [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)] \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i. \quad (73)$$



Again, if  $\nabla_{\mathcal{D}}^2$  cannot be evaluated exactly, then an ensemble-based approximation can be adopted, which will be discussed later. Assembling the results from Eqs. 67–73

into Eq. 65, we derive the following ensemble-based model update formula:

$$\begin{aligned} \mathbf{m}_j^{i+1} &= \mathbf{m}_j^i + \mathbf{S}_{\mathcal{I}}^i \left[ \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) \right]^{-1} \left( \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right)^T \left\{ \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \left[ \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) \right]^{-1} \left( \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right)^T + \gamma^i \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \left[ \mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) \right]^{-1} \left( \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right)^T \right\}^{-1} \times \\ &\quad \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \left[ \mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) \right]^{-1} \left( \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right)^T \nabla_{\mathcal{D}} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i) \right] \\ &= \mathbf{m}_j^i + \mathbf{S}_{\mathcal{I}}^i \left[ \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) \right]^{-1} \left\{ \left[ \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) \right]^{-1} + \gamma^i \left[ \mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) \right]^{-1} \right\}^{-1} \left[ \mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) \right]^{-1} \left( \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right)^T \nabla_{\mathcal{D}} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i) \right] \\ &= \mathbf{m}_j^i + \mathbf{S}_{\mathcal{I}}^i \left[ \mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) + \gamma^i \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) \right]^{-1} \left( \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right)^T \nabla_{\mathcal{D}} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i) \right] \\ &= \mathbf{m}_j^i + \tilde{\mathbf{K}}_j^i \nabla_{\mathcal{D}} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i) \right]. \end{aligned} \tag{74}$$

$$\tilde{\mathbf{K}}_j^i \equiv \mathbf{S}_{\mathcal{I}}^i \left[ \mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) + \gamma^i \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) \right]^{-1} \left( \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right)^T. \tag{75}$$

In Eq. 74, if the analytic form of the gradient  $\nabla_{\mathcal{D}}$  is not available, then an ensemble-based approximation strategy can be adopted for its computation, as will be discussed later. Equation 75 defines an effective Kalman-gain-like matrix  $\tilde{\mathbf{K}}_j^i$  involved in the model update formula (74). For the purpose of calculating  $\tilde{\mathbf{K}}_j^i$ , the involved square-root matrices  $\mathbf{S}_{\mathcal{I}}^i$  and  $\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i$  are common to all ensemble members. The differences among ensemble members reside in the matrices  $\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)$  and  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$ , which are both in the dimension of  $N_e \times N_e$ . Assuming that  $\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)$  and  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$  vary over all ensemble members, then the storage costs of  $\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)$  and  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$  are in the order of  $N_e^3$ , which is normally affordable in practice, with the typical ensemble size  $N_e$  being in the order of  $10^2$ .

Below we show that the original IES algorithm 3 is a special case of the more general formula in Eq. 74. To this end, let the distance metric  $\mathcal{D}$  and the regularization operator  $\mathcal{R}$  be some quadratic mappings as defined in Eqs. 11 and 12. In addition, let both the transform operators  $\mathcal{T}$  and  $\mathcal{S}$  be identity, such that

$$\begin{aligned} \mathbf{S}_{\mathcal{S}}^i &= \mathbf{S}_{\mathcal{I}}^i; \\ \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i &= \mathbf{S}_{\mathbf{g}}^i; \\ \nabla_{\mathcal{D}} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i) \right] &= \mathbf{C}_d^{-1} (\mathbf{d}_j - \mathbf{g}(\mathbf{m}_j^i)); \\ \mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j) &= \left( \mathbf{S}_{\mathbf{g}}^i \right)^T \mathbf{C}_d^{-1} \mathbf{S}_{\mathbf{g}}^i; \\ \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) &= \left( \mathbf{S}_{\mathcal{I}}^i \right)^T \left( \mathbf{C}_m^i \right)^{-1} \mathbf{S}_{\mathcal{I}}^i = \mathbf{I}_{N_e}, \end{aligned} \tag{76}$$

where  $\mathbf{I}_{N_e}$  is the identity matrix in the dimension of  $N_e \times N_e$ . As a result, the update formula (74) is reduced to

$$\mathbf{m}_j^{i+1} = \mathbf{m}_j^i + \mathbf{S}_{\mathcal{I}}^i \left[ \left( \mathbf{S}_{\mathbf{g}}^i \right)^T \mathbf{C}_d^{-1} \mathbf{S}_{\mathbf{g}}^i + \gamma^i \mathbf{I}_{N_e} \right]^{-1} \left( \mathbf{S}_{\mathbf{g}}^i \right)^T \mathbf{C}_d^{-1} (\mathbf{d}_j - \mathbf{g}(\mathbf{m}_j^i)). \tag{77}$$

According to Eq. 49, one has

$$\left[ \left( \mathbf{S}_{\mathbf{g}}^i \right)^T \mathbf{C}_d^{-1} \mathbf{S}_{\mathbf{g}}^i + \gamma^i \mathbf{I}_{N_e} \right]^{-1} \left( \mathbf{S}_{\mathbf{g}}^i \right)^T \mathbf{C}_d^{-1} = \left( \mathbf{S}_{\mathbf{g}}^i \right)^T \left( \mathbf{S}_{\mathbf{g}}^i \left( \mathbf{S}_{\mathbf{g}}^i \right)^T + \gamma^i \mathbf{C}_d \right)^{-1}. \tag{78}$$

Therefore, Eq. 77 is equivalent to

$$\mathbf{m}_j^{i+1} = \mathbf{m}_j^i + \mathbf{S}_{\mathcal{I}}^i \left( \mathbf{S}_{\mathbf{g}}^i \right)^T \left( \mathbf{S}_{\mathbf{g}}^i \left( \mathbf{S}_{\mathbf{g}}^i \right)^T + \gamma^i \mathbf{C}_d \right)^{-1} (\mathbf{d}_j - \mathbf{g}(\mathbf{m}_j^i)), \tag{79}$$

which is exactly the same as the update formula induced by Eqs. 3–7.

## Appendix B: Approximations to the products between gradient or Hessian and certain ensemble-induced square-root matrices

Here, we consider ensemble-based approximations to the term  $\left( \mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i \right)^T \nabla_{\mathcal{D}} \left[ \mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i) \right]$  in Eq. 74, the term  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$  defined in Eq. 70, and the term  $\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)$  defined in Eq. 73, when the analytic forms of  $\nabla_{\mathcal{D}}$ ,  $\nabla_{\mathcal{R}}^2$  and  $\nabla_{\mathcal{D}}^2$  are not available.

Similar to Eq. 68, by applying the first order Taylor approximation (45), one has

$$\begin{aligned}
 & (\mathbf{S}_{\mathcal{T} \circ \mathbf{g}}^i)^T \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)] \\
 &= \frac{1}{\sqrt{N_e - 1}} [\mathcal{T} \circ \mathbf{g}(\mathbf{m}_1^i) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i), \dots, \mathcal{T} \circ \mathbf{g}(\mathbf{m}_{N_e}^i) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)]^T \times \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)] \\
 &= \frac{1}{\sqrt{N_e - 1}} [(\mathcal{T} \circ \mathbf{g}(\mathbf{m}_1^i) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)) + (\mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)), \dots, \\
 &\quad (\mathcal{T} \circ \mathbf{g}(\mathbf{m}_{N_e}^i) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)) + (\mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i))]^T \times \nabla_{\mathcal{D}} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)] \\
 &\approx \frac{1}{\sqrt{N_e - 1}} [(\mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)) - \mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_1^i))) + \\
 &\quad (\mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)) - \mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i))), \\
 &\quad \dots, \\
 &\quad (\mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)) - \mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_{N_e}^i))) + \\
 &\quad (\mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)) - \mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_j^i)))] \\
 &= \frac{\mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i))}{\sqrt{N_e - 1}} \mathbf{1}_{N_e} - \frac{1}{\sqrt{N_e - 1}} [\mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_1^i)), \dots, \mathcal{D}(\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_{N_e}^i))]^T, \tag{80}
 \end{aligned}$$

where  $\mathbf{1}_{N_e}$  is a  $N_e$ -long vector whose elements are all equal to 1.

Next, consider the ensemble approximation to  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$ . First, let us evaluate

$$\begin{aligned}
 \mathbf{p}_1 &= (\nabla_{\mathcal{R}}^2 [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)]) \mathbf{S}_c^i \\
 &= \frac{1}{\sqrt{N_e - 1}} (\nabla_{\mathcal{R}}^2 [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)]) \\
 &\quad [S(\mathbf{m}_1^i) - S(\mathbf{m}_c^i), \dots, S(\mathbf{m}_{N_e}^i) - S(\mathbf{m}_c^i)] \\
 &\approx \frac{1}{\sqrt{N_e - 1}} [\nabla_{\mathcal{R}} [S(\mathbf{m}_1^i) - S(\mathbf{m}_j^i)] - \nabla_{\mathcal{R}} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)], \dots, \\
 &\quad \nabla_{\mathcal{R}} [S(\mathbf{m}_{N_e}^i) - S(\mathbf{m}_j^i)] - \nabla_{\mathcal{R}} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)]]. \tag{81}
 \end{aligned}$$

Then we have

$$\begin{aligned}
 \mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i) &= (\mathbf{S}_c^i)^T \mathbf{p}_1 \\
 &\approx \frac{1}{N_e - 1} [S(\mathbf{m}_1^i) - S(\mathbf{m}_c^i), \dots, S(\mathbf{m}_{N_e}^i) - S(\mathbf{m}_c^i)]^T \\
 &\quad \times [\nabla_{\mathcal{R}} [S(\mathbf{m}_1^i) - S(\mathbf{m}_j^i)] - \nabla_{\mathcal{R}} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)], \dots, \\
 &\quad \nabla_{\mathcal{R}} [S(\mathbf{m}_{N_e}^i) - S(\mathbf{m}_j^i)] - \nabla_{\mathcal{R}} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)]]. \tag{82}
 \end{aligned}$$

The element of  $\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)$  on the  $k$ th row ( $k = 1, 2, \dots, N_e$ ) and the  $\ell$ th column ( $\ell = 1, 2, \dots, N_e$ ), denoted by  $[\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)]_{k,\ell}$ , is then given by

$$\begin{aligned}
 [\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)]_{k,\ell} &= \frac{1}{N_e - 1} (S(\mathbf{m}_k^i) - S(\mathbf{m}_c^i))^T \{ \nabla_{\mathcal{R}} [S(\mathbf{m}_\ell^i) - S(\mathbf{m}_j^i)] - \nabla_{\mathcal{R}} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)] \} \\
 &\approx \frac{1}{N_e - 1} \{ \mathcal{R} [S(\mathbf{m}_k^i) + S(\mathbf{m}_\ell^i) - S(\mathbf{m}_j^i) - S(\mathbf{m}_c^i)] + \mathcal{R} [S(\mathbf{m}_c^i) - S(\mathbf{m}_j^i)] \\
 &\quad - \mathcal{R} [S(\mathbf{m}_k^i) - S(\mathbf{m}_j^i)] - \mathcal{R} [S(\mathbf{m}_\ell^i) - S(\mathbf{m}_j^i)] \}. \tag{83}
 \end{aligned}$$

Clearly, one has  $[\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)]_{k,\ell} = [\mathbf{M}_{\mathcal{R}}^i(\mathbf{m}_j^i)]_{\ell,k}$ , meaning that the ensemble approximation leads to a symmetric matrix.

In a similar way, the elements of the ensemble approximation to the term  $\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)$  can be obtained as follows:

$$\begin{aligned}
 [\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)]_{k,\ell} &\approx \frac{1}{N_e - 1} \{ \mathcal{D} [\mathcal{T}(\mathbf{d}_j) + \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_k^i) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_\ell^i)] + \mathcal{D} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_c^i)] \\
 &\quad - \mathcal{D} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_k^i)] - \mathcal{D} [\mathcal{T}(\mathbf{d}_j) - \mathcal{T} \circ \mathbf{g}(\mathbf{m}_\ell^i)] \}. \tag{84}
 \end{aligned}$$

Again, the ensemble approximation leads to a symmetric matrix, since  $[\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)]_{k,\ell} = [\mathbf{M}_{\mathcal{D}}^i(\mathbf{d}_j)]_{\ell,k}$ .

### Appendix C: Analytic forms of gradient and Hessian with respect to the $\ell_p^q$ metric

In the sequel, we derive the analytic forms of gradient and Hessian of the distance metric  $\|\mathbf{B}\mathbf{x}\|_p^q$ , where  $\mathbf{x}$  is an  $m_x$ -dimensional vector, and  $\mathbf{B}$  a matrix in the dimension of  $N_b \times m_x$ . As a result, the product  $\mathbf{B}\mathbf{x}$  leads to a  $N_b$ -dimensional vector, while the meaning of  $\|\mathbf{B}\mathbf{x}\|_p^q$  is to raise the  $\ell_p$  norm of  $\mathbf{B}\mathbf{x}$  to the  $q$ th power. In short, we call  $\|\mathbf{B}\mathbf{x}\|_p^q$  a  $\ell_p^q$  metric hereafter.

By definition, we have the  $\ell_p^q$  metric

$$\|\mathbf{B}\mathbf{x}\|_p^q = \left( \sum_{e=1}^{N_b} |(\mathbf{B}\mathbf{x})_e|^p \right)^{q/p}; \tag{85}$$

$$(\mathbf{B}\mathbf{x})_e \equiv \sum_{f=1}^{m_x} B_{e,f} x_f, \tag{86}$$

where  $(\mathbf{B}\mathbf{x})_e$  stands for the  $e$ th element of the vector  $\mathbf{B}\mathbf{x}$ ,  $x_f$  for the  $f$ th element of the vector  $\mathbf{x}$  ( $f = 1, 2, \dots, m_x$ ), and  $B_{e,f}$  for the element of the matrix  $\mathbf{B}$  on the  $e$ th row and the  $f$ th column.

With some algebra, one has

$$\begin{aligned} \frac{\|\mathbf{B}\mathbf{x}\|_p^q}{\partial x_k} &= \frac{\partial}{\partial x_k} \left( \sum_{e=1}^{N_b} \left| \sum_{f=1}^{m_x} B_{e,f} x_f \right|^p \right)^{q/p} \\ &= \frac{q}{p} \left( \sum_{e=1}^{N_b} \left| \sum_{f=1}^{m_x} B_{e,f} x_f \right|^p \right)^{(q-p)/p} \left[ \sum_{e=1}^{N_b} p \left| \sum_{f=1}^{m_x} B_{e,f} x_f \right|^{p-1} \frac{\partial}{\partial x_k} \left| \sum_{f=1}^{m_x} B_{e,f} x_f \right| \right] \\ &= q \left( \sum_{e=1}^{N_b} \left| \sum_{f=1}^{m_x} B_{e,f} x_f \right|^p \right)^{(q-p)/p} \left[ \sum_{e=1}^{N_b} \left| \sum_{f=1}^{m_x} B_{e,f} x_f \right|^{p-2} \left( \sum_{f=1}^{m_x} B_{e,f} x_f \right) B_{e,k} \right] \\ &= q \|\mathbf{B}\mathbf{x}\|_p^{q-p} \left[ \sum_{e=1}^{N_b} |(\mathbf{B}\mathbf{x})_e|^{p-2} ((\mathbf{B}\mathbf{x})_e) B_{e,k} \right]. \end{aligned} \tag{87}$$

Note that to derive the result in the third line of Eq. 87 from the second one, we have used the result that  $d|x|/dx = \text{sgn}(x) = x/|x|$  (with  $\text{sgn}$  being the sign function), while ignoring the singularity of the derivative at  $x = 0$ . In the context of ensemble-based data assimilation, this way of handling the singularity at  $x = 0$  may work well in general, since we are typically dealing with gradient and Hessian evaluated at local points away from 0, due to the deviations of ensemble members from the ensemble mean. For example, see the formula in Eq. 16.

Based on Eq. 87, one has

$$\begin{aligned} \nabla_{\|\mathbf{B}\mathbf{x}\|_p^q}(\mathbf{x}) &\equiv \frac{\partial \|\mathbf{B}\mathbf{x}\|_p^q}{\partial \mathbf{x}} \\ &= q \|\mathbf{B}\mathbf{x}\|_p^{q-p} \mathbf{B}^T \mathbf{a}; \tag{88} \\ \mathbf{a} &\equiv |\mathbf{B}\mathbf{x}|^{\wedge(p-2)} \odot (\mathbf{B}\mathbf{x}). \end{aligned} \tag{89}$$

In Eq. 89, the operator  $\wedge$  raises the elements of a vector to a certain power, e.g., for a (column) vector  $\mathbf{y} = [y_1, y_2, \dots]^T$ , one has  $|\mathbf{y}|^{\wedge p} = [|y_1|^p, |y_2|^p, \dots]^T$ . On the other hand, the operator  $\odot$  stands for element-wise product (i.e., Hadamard product), such that for  $\mathbf{x} = [x_1, x_2, \dots]^T$  and  $\mathbf{y} = [y_1, y_2, \dots]^T$ , one has  $\mathbf{x} \odot \mathbf{y} = [x_1 y_1, x_2 y_2, \dots]^T$ .

As special cases, if  $p = q = 2$  or  $p = q = 1$ , then Eqs. 88 and 89 lead to

$$\nabla_{\|\mathbf{B}\mathbf{x}\|_2^2}(\mathbf{x}) = 2\mathbf{B}^T \mathbf{B}\mathbf{x},$$

or

$$\nabla_{\|\mathbf{B}\mathbf{x}\|_1}(\mathbf{x}) = \mathbf{B}^T \text{sgn}(\mathbf{B}\mathbf{x}),$$

where  $\text{sgn}$  denotes the sign function applied element-wise to its input.

In a similar way, we can proceed to compute the Hessian of  $\|\mathbf{B}\mathbf{x}\|_p^q$ . With some algebra, we obtain

$$\begin{aligned} \frac{\|\mathbf{B}\mathbf{x}\|_p^q}{\partial x_k \partial x_s} &= q(q-p) \|\mathbf{B}\mathbf{x}\|_p^{q-2p} \left[ \sum_{e=1}^{N_b} |(\mathbf{B}\mathbf{x})_e|^{p-2} ((\mathbf{B}\mathbf{x})_e) B_{e,k} \right] \\ &\quad \left[ \sum_{e=1}^{N_b} |(\mathbf{B}\mathbf{x})_e|^{p-2} ((\mathbf{B}\mathbf{x})_e) B_{e,s} \right] + q(p-1) \|\mathbf{B}\mathbf{x}\|_p^{q-p} \\ &\quad \left[ \sum_{e=1}^{N_b} B_{e,k} B_{e,s} |(\mathbf{B}\mathbf{x})_e|^{p-2} \right]. \end{aligned} \tag{90}$$

As a result, we have

$$\begin{aligned} \nabla_{\|\mathbf{B}\mathbf{x}\|_p^q}^2(\mathbf{x}) &= q(q-p) \|\mathbf{B}\mathbf{x}\|_p^{q-2p} \mathbf{B}^T \mathbf{a} \mathbf{a}^T \mathbf{B} + q(p-1) \|\mathbf{B}\mathbf{x}\|_p^{q-p} (\mathbf{B} \odot \mathbf{C})^T (\mathbf{B} \odot \mathbf{C}); \tag{91} \\ \mathbf{C} &\equiv |\mathbf{B}\mathbf{x}|^{\wedge(p/2-1)} \mathbf{1}_{m_x}^T, \end{aligned} \tag{92}$$

where  $\mathbf{1}_{m_x}$  stands for a  $m_x$ -dimensional vector whose elements are all equal to 1. As is evident in Eq. 91,  $\nabla_{\|\mathbf{B}\mathbf{x}\|_p^q}^2(\mathbf{x})$  is symmetric. Interestingly, if  $0 < q = p < 1$ , one can see that  $\nabla_{\|\mathbf{B}\mathbf{x}\|_p^q}^2(\mathbf{x})$  is negative semi-definite, reflecting the fact that the metric  $\|\mathbf{B}\mathbf{x}\|_p^q$  is non-convex in this case.

Again, if  $p = q = 2$  or  $p = q = 1$ , then Eqs. 91 and 92 lead to

$$\nabla_{\|\mathbf{B}\mathbf{x}\|_2^2}^2(\mathbf{x}) = 2\mathbf{B}^T \mathbf{B},$$

or

$$\nabla_{\|\mathbf{B}\mathbf{x}\|_1}^2(\mathbf{x}) = \mathbf{0}, \mathbf{B}\mathbf{x} \neq \mathbf{0}.$$

The latter result implies that the Hessian of  $\|\mathbf{B}\mathbf{x}\|_1$  vanishes almost everywhere (except at  $\mathbf{B}\mathbf{x} = \mathbf{0}$ ).

### Appendix D: Gradient and Hessian of a mixture of regularization operators

Here we aim to evaluate the gradient and Hessian of a mixture of  $K_{mix}$  regularization operators in the transformed model space, in the form of

$$\sum_{k=1}^{K_{mix}} w_k \mathcal{R}_k(\mathcal{S}_k(\mathbf{x})), \tag{93}$$

where  $w_k$  is a scalar, representing the weight associated with the  $k$ th regularization operator  $\mathcal{R}_k$ , which acts on the transformed model variables  $\mathcal{S}_k(\mathbf{x})$ , obtained by applying the transform operator  $\mathcal{S}_k$  to the model variables  $\mathbf{x}$ .

For ease of comprehension, in the sequel we adopt the method of induction. To start, consider the case with a single regularization operator  $\mathcal{R}$ , such that the regularization term is simply in the form of  $\mathcal{R}(\mathcal{S}(\mathbf{x}))$ . In this case, we have

$$\nabla_{\mathcal{R}}(\mathbf{x}) \equiv \frac{\partial \mathcal{R}(\mathcal{S}(\mathbf{x}))}{\partial \mathbf{x}} = \frac{\partial \mathcal{S}}{\partial \mathbf{x}} \frac{\partial \mathcal{R}(\mathcal{S})}{\partial \mathcal{S}} = \nabla_{\mathcal{S}}(\mathbf{x}) \nabla_{\mathcal{R}}(\mathcal{S}). \tag{94}$$

To get the Hessian, we have

$$\begin{aligned} \nabla_{\mathcal{R}}^2(\mathbf{x}) &= \frac{\partial}{\partial \mathbf{x}} \left( \frac{\partial \mathcal{S}}{\partial \mathbf{x}} \frac{\partial \mathcal{R}(\mathcal{S})}{\partial \mathcal{S}} \right) \\ &= \frac{\partial}{\partial \mathbf{x}} \left( \frac{\partial \mathcal{S}}{\partial \mathbf{x}} \right)^T \frac{\partial \mathcal{R}(\mathcal{S})}{\partial \mathcal{S}} + \frac{\partial}{\partial \mathbf{x}} \left( \frac{\partial \mathcal{R}(\mathcal{S})}{\partial \mathcal{S}} \right) \left( \frac{\partial \mathcal{S}}{\partial \mathbf{x}} \right)^T \\ &\approx \frac{\partial \mathcal{S}}{\partial \mathbf{x}} \frac{\partial}{\partial \mathcal{S}} \left( \frac{\partial \mathcal{R}(\mathcal{S})}{\partial \mathcal{S}} \right) \left( \frac{\partial \mathcal{S}}{\partial \mathbf{x}} \right)^T \\ &= \nabla_{\mathcal{S}}(\mathbf{x}) \nabla_{\mathcal{R}}^2(\mathcal{S}) (\nabla_{\mathcal{S}}(\mathbf{x}))^T. \end{aligned} \tag{95}$$

To derive the third line of Eq. 95, we ignore the Hessian of the transform operator  $\mathcal{S}$  with respect to  $\mathbf{x}$ , in light of the strategy of first-order Taylor approximation in Eq. 55.

Now let us consider the case of a mixture of two regularization operators, in terms of  $w_1 \mathcal{R}_1(\mathcal{S}_1(\mathbf{x})) + w_2 \mathcal{R}_2(\mathcal{S}_2(\mathbf{x}))$ . In this case, we define

$$\begin{aligned} \mathcal{R}(\mathcal{S}(\mathbf{x})) &\equiv w_1 \mathcal{R}_1(\mathcal{S}_1(\mathbf{x})) + w_2 \mathcal{R}_2(\mathcal{S}_2(\mathbf{x})) \\ &= \mathbf{w}^T \tilde{\mathcal{R}}(\mathcal{S}(\mathbf{x})); \end{aligned} \tag{96}$$

$$\mathcal{S}(\mathbf{x}) \equiv \left[ (\mathcal{S}_1(\mathbf{x}))^T, (\mathcal{S}_2(\mathbf{x}))^T \right]^T; \tag{97}$$

$$\mathbf{w} = [w_1, w_2]^T, \tag{98}$$

$$\tilde{\mathcal{R}}(\mathcal{S}(\mathbf{x})) = \left[ (\mathcal{R}_1(\mathcal{S}_1(\mathbf{x})))^T, (\mathcal{R}_2(\mathcal{S}_2(\mathbf{x})))^T \right]^T. \tag{99}$$

As a result, we obtain

$$\nabla_{\mathcal{R}}(\mathbf{x}) \equiv \frac{\partial \mathcal{R}(\mathcal{S}(\mathbf{x}))}{\partial \mathbf{x}} = \frac{\partial \mathcal{S}}{\partial \mathbf{x}} \frac{\partial \tilde{\mathcal{R}}(\mathcal{S})}{\partial \mathcal{S}} \mathbf{w} = \nabla_{\mathcal{S}}(\mathbf{x}) \nabla_{\tilde{\mathcal{R}}}(\mathcal{S}) \mathbf{w}; \tag{100}$$

$$\nabla_{\mathcal{S}}(\mathbf{x}) = [\nabla_{\mathcal{S}_1}(\mathbf{x}), \nabla_{\mathcal{S}_2}(\mathbf{x})]; \tag{101}$$

$$\begin{aligned} \nabla_{\tilde{\mathcal{R}}}(\mathcal{S}) &= \begin{pmatrix} \nabla_{\mathcal{R}_1}(\mathcal{S}_1) & \mathbf{0} \\ \mathbf{0} & \nabla_{\mathcal{R}_2}(\mathcal{S}_2) \end{pmatrix}. \end{aligned} \tag{102}$$

$$\tag{103}$$

Therefore, we have the gradient of  $\mathcal{R}$  with respect to  $\mathcal{S}$ , in terms of

$$\nabla_{\mathcal{R}}(\mathcal{S}) \equiv \nabla_{\tilde{\mathcal{R}}}(\mathcal{S}) \mathbf{w} = \left[ w_1 (\nabla_{\mathcal{R}_1}(\mathcal{S}_1))^T, w_2 (\nabla_{\mathcal{R}_2}(\mathcal{S}_2))^T \right]^T. \tag{104}$$

In addition, we have

$$\begin{aligned} \nabla_{\mathcal{R}}^2(\mathbf{x}) &= \frac{\partial}{\partial \mathbf{x}} (\nabla_{\mathcal{S}}(\mathbf{x}) \nabla_{\tilde{\mathcal{R}}}(\mathcal{S}) \mathbf{w}) \\ &\approx \frac{\partial}{\partial \mathbf{x}} (\nabla_{\tilde{\mathcal{R}}}(\mathcal{S}) \mathbf{w}) (\nabla_{\mathcal{S}}(\mathbf{x}))^T \end{aligned} \tag{105}$$

$$= \nabla_{\mathcal{S}}(\mathbf{x}) \left[ \frac{\partial}{\partial \mathcal{S}} (\nabla_{\tilde{\mathcal{R}}}(\mathcal{S}) \mathbf{w}) \right] (\nabla_{\mathcal{S}}(\mathbf{x}))^T. \tag{106}$$

Hence the Hessian of  $\mathcal{R}$  with respect to  $\mathcal{S}$  is given by

$$\begin{aligned} \nabla_{\mathcal{R}}^2(\mathcal{S}) &\equiv \frac{\partial}{\partial \mathcal{S}} (\nabla_{\tilde{\mathcal{R}}}(\mathcal{S}) \mathbf{w}) \\ &= \begin{pmatrix} w_1 \nabla_{\mathcal{R}_1}^2(\mathcal{S}_1) & \mathbf{0} \\ \mathbf{0} & w_2 \nabla_{\mathcal{R}_2}^2(\mathcal{S}_2) \end{pmatrix}. \end{aligned} \tag{107}$$

To obtain (107), we have used the fact that  $\partial \nabla_{\mathcal{R}_i}(\mathcal{S}_i) / \partial \mathcal{S}_j = \mathbf{0}$ , for  $i \neq j$ ,  $i, j \in \{1, 2\}$ .

Similarly, for a mixture of  $K_{mix}$  regularization operators, i.e.,

$$\mathcal{R}(\mathcal{S}(\mathbf{x})) \equiv \sum_{k=1}^{K_{mix}} w_k \mathcal{R}_k(\mathcal{S}_k(\mathbf{x})),$$

one has its gradient and Hessian in the transformed model space given by

$$S(\mathbf{x}) \equiv \left[ (S_1(\mathbf{x}))^T, (S_2(\mathbf{x}))^T, \dots, (S_{K_{mix}}(\mathbf{x}))^T \right]^T; \tag{108}$$

$$\nabla_{\mathcal{R}}(S) = \left[ w_1 (\nabla_{\mathcal{R}_1}(S_1))^T, w_2 (\nabla_{\mathcal{R}_2}(S_2))^T, \dots, w_{K_{mix}} (\nabla_{\mathcal{R}_{K_{mix}}}(S_{K_{mix}}))^T \right]^T; \tag{109}$$

$$\nabla_{\mathcal{R}}^2(S) = \begin{pmatrix} w_1 \nabla_{\mathcal{R}_1}^2(S_1) & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & w_2 \nabla_{\mathcal{R}_2}^2(S_2) & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & w_{K_{mix}} \nabla_{\mathcal{R}_{K_{mix}}}^2(S_{K_{mix}}) \end{pmatrix}. \tag{110}$$

### Appendix E: Additional results in a 5-spots example

#### The reservoir model

Here we present some additional numerical results obtained by applying a subset of  $\ell_p^q$ -GIES algorithms to a 2D 5-spots example, wherein the numerical reservoir model is in the dimension of  $50 \times 50$  (in the unit of gridblocks), with oil, water and gas phases. There are four producers (labeled as P1 – P4) on the corners of the model, and one injector (labeled as I1) in the center, as indicated in Fig. 11. The locations of these wells, in terms of Cartesian coordinates  $(x, y)$ , are as follows: P1 at (2, 2); P2 at (49, 2); P3 at (2, 49); P4 at (49, 49) and I1 at (25, 25).

The parameters to be estimated consist of x-dimensional permeability (PERMX) and porosity (PORO) on all reservoir gridblocks. Initial ensembles for PERMX and PORO (with 100 ensemble members) are generated through sequential Gaussian simulation. Figure 11 shows the PERMX and PORO maps in the reference model, which is used to generate production data every 30 days, for a total period of 1500 days. One of the producers, P4, is shut in during history matching, therefore the production data consist of well oil/water/gas production rates (WOPR/WWPR/WGPR) from the other 3 producers, and well bottom-hole pressures (WBHP) from wells P1 – P3 and I1 every 30 days. The number of production data used for history matching is 650 in total. The observed production data are contaminated by certain zero-mean Gaussian white noise. For WOPR/WWPR/WGPR data, the standard deviations (STDs) of observation noise are either 10% of their magnitudes, or set to  $10^{-6}$  if the observed production data (e.g., WWPR) are equal to zero; whereas for WBHP, their STDs are 1 bar.

The  $\ell_p^q$ -GIES algorithms adopted for this case study are constructed based on the following considerations. As Figure 11 indicates, there are also spatial patterns in the reservoir models that can be exploited by the algorithms, e.g., through the calculations of the first-order variations of PERMX and PORO maps, similar to the previous

channelized reservoir characterization problem. On the other hand, histograms of PERMX and PORO in the current case study do not provide particularly useful information of spatial patterns. As such, we choose to construct the  $\ell_p^q$ -GIES algorithms by solving the GMAC problem with the following cost function:

$$C_j^{i+1} = \frac{1}{2} (\mathbf{d}_j - \mathbf{g}(\mathbf{m}_j^{i+1}))^T \mathbf{C}_d^{-1} (\mathbf{d}_j - \mathbf{g}(\mathbf{m}_j^{i+1})) + \gamma^i \mathcal{R} [S(\mathbf{m}_j^{i+1}) - S(\mathbf{m}_j^i)]; \tag{111}$$

$$\mathcal{R} [S(\mathbf{m}_j^{i+1}) - S(\mathbf{m}_j^i)] = \frac{1}{2} \left[ w_1 (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)^T (\mathbf{C}_m^i)^{-1} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i) + w_2 \|S_V(\mathbf{m}_j^{i+1}) - S_V(\mathbf{m}_j^i)\|_2^2 + w_3 \|S_V(\mathbf{m}_j^{i+1}) - S_V(\mathbf{m}_j^i)\|_1^2 \right], \tag{112}$$

where  $S_V$  is a transform operator similar to that in the channelized reservoir characterization problem, but applied to both PERMX and PORO maps; and the weights  $w_i$  ( $i = 1, 2, 3$ ) are determined in a way similar to that in Eq. 43, which leads to the scalar coefficients  $\alpha_i$  ( $i = 1, 2, 3$ ) in Table 4.

Similar to the situation in the channelized reservoir characterization problem, we adopt 3-bit binary codes to refer to the  $\ell_p^q$ -GIES algorithms derived from Eq. 111, which leads to 7 algorithms after excluding the one with the code 000 (i.e., no regularization). The characteristics of these 7 algorithms are the same as those summarized in Table 2, except that here only three individual regularization terms, namely,  $(\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)^T (\mathbf{C}_m^i)^{-1} (\mathbf{m}_j^{i+1} - \mathbf{m}_j^i)$ ,  $\|S_V(\mathbf{m}_j^{i+1}) - S_V(\mathbf{m}_j^i)\|_2^2$  and  $\|S_V(\mathbf{m}_j^{i+1}) - S_V(\mathbf{m}_j^i)\|_1^2$ , are adopted. All these  $\ell_p^q$ -GIES algorithms are equipped with an automatic and adaptive localization scheme, RndShfl-GC, of [31] during history matching, and are run with 10 iteration steps.

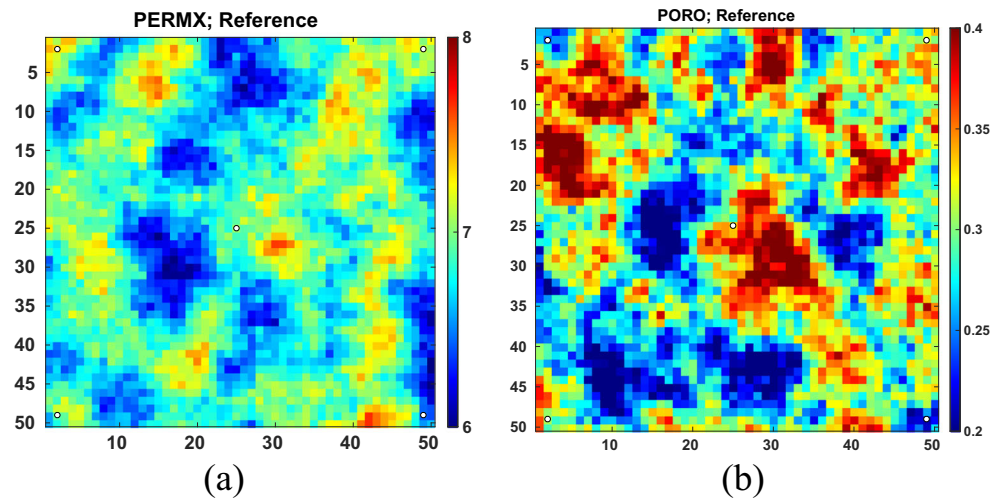
In the example here, we use root mean square error (RMSE) as a measure to cross-validate the history matching performance. Given an  $m$ -dimensional reference model  $\mathbf{m}^{ref}$  and an ensemble  $\mathcal{M}^i = \{\mathbf{m}_j^i\}_{j=1}^{N_e}$  of reservoir models at the  $i$ th iteration step, we compute an ensemble  $\Omega(\mathcal{M}^i)$  of RMSE as follows:

$$\Omega(\mathcal{M}^i) \equiv \left\{ \zeta_j^i | \zeta_j^i = \frac{\|\mathbf{m}_j^i - \mathbf{m}^{ref}\|_2^2}{\sqrt{m}} \right\}_{j=1}^{N_e}. \tag{113}$$

Table 4 reports data mismatch and RMSE values (in the form of mean  $\pm$  STD) with respect to the final ensembles of the 7  $\ell_p^q$ -GIES algorithms. The RMSE values are listed for both PERMX and PORO. For reference, data mismatch, RMSE of PORO and RMSE of PERMX with respect to the initial ensemble are  $(1.7376 \pm 4.3294) \times 10^4$ ,  $0.0644 \pm 0.0037$  and  $0.4367 \pm 0.0263$ , respectively. For performance assessment, we adopt the average RMSE over PERMX



**Fig. 11** Reference PERMX and PORO maps used in the synthetic case study, with well positions marked by white ellipses (four producers on the corners and one injector in the center). **a** Reference PERMX map. **b** Reference PORO map



and PORO (with equal weights). Following this setting, the original IES (code 100) is ranked as the 6th, meaning that there are a few other  $\ell_p^q$ -GIES algorithms that outperform the original IES. As such, one can draw a conclusion similar to that in the previous channelized reservoir characterization problem, that is, it is possible to design through the GIES framework new ensemble history matching algorithms that may perform better than the original IES algorithm in certain circumstances.

In the current case study, except for the  $\ell_p^q$ -GIES algorithm with the code 001 (for convenience, hereafter we simply call it algorithm 001, and the same custom will be applied to other algorithms), the other algorithms result in very similar RMSE values for the estimated PORO maps. On the other hand, the differences among the RMSE values of PERMX are more substantial. In particular, for algorithm 001, although its mean data mismatch value is higher, its RMSE values for both PORO and PERMX are the lowest among the tested algorithms.

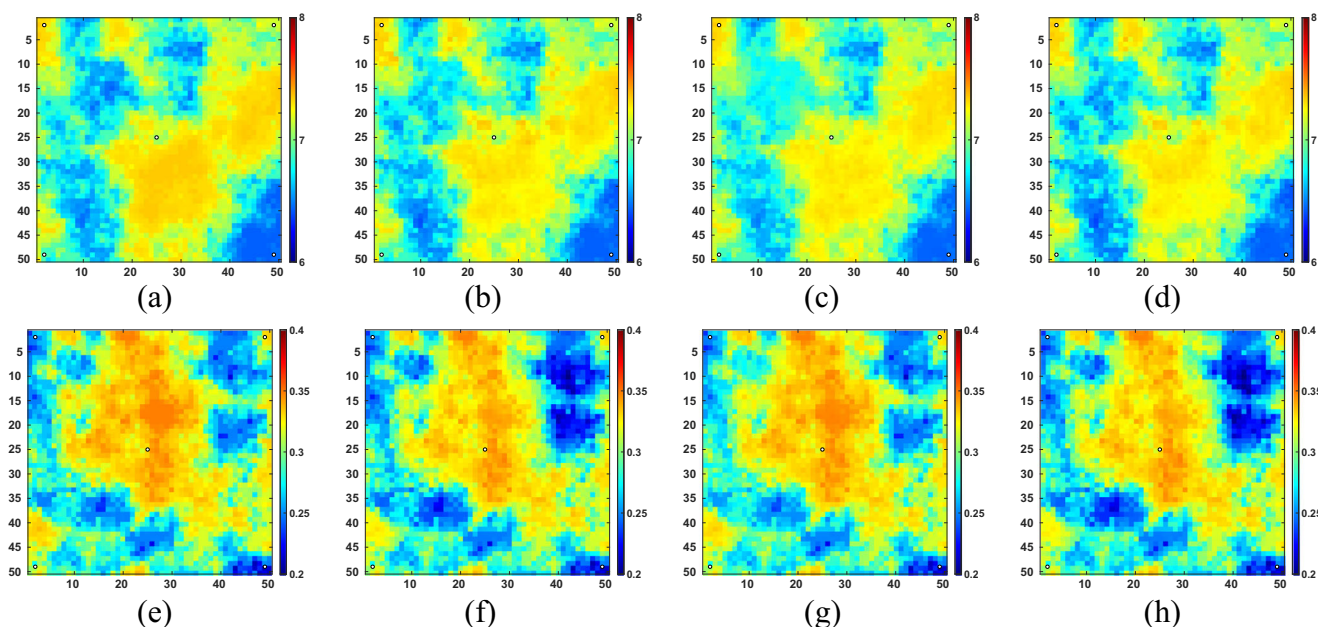
To limit the length of the current work, we do not proceed to present the full numerical results. Instead,

we focus on inspecting the impacts of a few  $\ell_p^q$ -GIES algorithms on model updates. To this end, Fig. 12 reports the PERMX and PORO maps of one initial reservoir model, and the corresponding maps of the final models obtained by algorithms 100 (i.e., the original IES), 001 and 010, respectively. In line with the results in Table 4, the PORO maps obtained by algorithms 100 and 010, respectively, are very similar. In comparison to the initial PORO map, a noticeable difference can be observed in the area around the coordinate (45, 15). On the other hand, the final PORO map obtained by algorithm 001 bears more structural differences from those of the other two algorithms. Meanwhile, compared to the initial PORO map, the final PORO map of algorithm 001 also exhibits some clear differences in the area around the coordinate (12, 7). A similar conclusion can be drawn for PERMX maps, especially if one compares the PERMX maps in the area around the coordinate (18, 12). As such, in this particular case study, it appears that, due to the use of the regularization term  $\|\mathcal{S}_V(\mathbf{m}_j^{i+1}) - \mathcal{S}_V(\mathbf{m}_j^i)\|_1^2$  with the  $\ell_1$  norm, algorithm 001 is able to produce flatter

**Table 4** Performance of  $\ell_p^q$ -GIES algorithms in terms of RMSE, which are evaluated with respect to the ensembles of reservoir models at the final iteration steps

Rank	Binary code	History-matching data mismatch (mean $\pm$ STD)	RMSE of PORO (mean $\pm$ STD)	RMSE of PERMX (mean $\pm$ STD)	Weights ( $\alpha_1, \alpha_2, \alpha_3$ )
1	001	1191.5353 $\pm$ 1455.8489	0.0619 $\pm$ 0.0035	0.4045 $\pm$ 0.025	(0, 0, 1)
2	010	502.441 $\pm$ 96.276	0.0637 $\pm$ 0.0033	0.4156 $\pm$ 0.0235	(0, 1, 0)
3	111	492.3271 $\pm$ 86.3171	0.0637 $\pm$ 0.0033	0.4194 $\pm$ 0.024	(0.4, 0.4, 0.2)
4	110	488.5942 $\pm$ 86.6626	0.0635 $\pm$ 0.0033	0.4202 $\pm$ 0.0239	(0.5, 0.5, 0)
5	011	548.0362 $\pm$ 314.1955	0.0636 $\pm$ 0.0033	0.4208 $\pm$ 0.0242	(0, 0.5, 0.5)
6	100	501.1523 $\pm$ 94.3388	0.0635 $\pm$ 0.0033	0.4211 $\pm$ 0.0244	(1, 0, 0)
7	101	498.7347 $\pm$ 92.2996	0.0637 $\pm$ 0.0033	0.4238 $\pm$ 0.0246	(0.5, 0, 0.5)

The  $\ell_p^q$ -GIES algorithms are listed in an ascending order of mean RMSE values. In particular, performance of the  $\ell_p^q$ -GIES algorithm corresponding to the original IES is highlighted (in red)



**Fig. 12** PERMX (top) and PORO (bottom) maps. The first column corresponds to the maps of one of the initial (Init.) reservoir models, whereas the second to the fourth columns are with respect to the final estimated (Est.) reservoirs, obtained by the  $\ell_p^q$ -GIES algorithms with

the codes 100 (i.e., the original IES), 001 and 010, respectively. **a** Init. PERMX. **b** Est. (100). **c** Est. (001). **d** Est. (010). **e** Init. PORO. **f** Est. (100). **g** Est. (001). **h** Est. (010)

regions, in which the estimated parameters (PERMX and PORO) exhibit less spatial variations. This property is less noticeable in algorithms 100 and 010, in which the distance is measured by the  $\ell_2$  norm instead.

**Acknowledgements** The author would like to thank two anonymous reviewers for their valuable and constructive suggestions. The author acknowledges financial support from the Research Council of Norway through the Petromaks-2 project DIGIRES (RCN no. 280473) and the industrial partners AkerBP, Wintershall DEA, Vår Energi, Petrobras, Equinor, Lundin and Neptune Energy, and would also like to thank Schlumberger for providing academic licenses to ECLIPSE<sup>®</sup>.

**Funding** Open access funding provided by NORCE Norwegian Research Centre AS.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Kay, S.M.: Fundamentals of statistical signal processing. vol 1: Estimation theory. Prentice Hall PTR (1993)
2. Evensen, G.: Data assimilation: The ensemble Kalman filter. Springer Science & Business Media (2009)
3. Kalnay, E.: Atmospheric modeling, data assimilation and predictability. Cambridge University Press (2002)
4. Oliver, D.S., Reynolds, A.C., Liu, N.: Inverse theory for petroleum reservoir characterization and history matching. Cambridge University Press (2008)
5. Tarantola, A.: Inverse problem theory and methods for model parameter estimation. SIAM (2005)
6. Engl, H.W., Hanke, M., Neubauer, A.: Regularization of inverse problems. Springer (2000)
7. Chen, Y., Oliver, D.: Levenberg-Marquardt forms of the iterative ensemble smoother for efficient history matching and uncertainty quantification. *Comput. Geosci.* **17**, 689–703 (2013)
8. Chen, Y., Oliver, D.S.: Ensemble randomized maximum likelihood method as an iterative ensemble smoother. *Math. Geosci.* **44**, 1–26 (2012)
9. Emerick, A.A.: Deterministic ensemble smoother with multiple data assimilation as an alternative for history-matching seismic data. *Comput. Geosci.* **22**(5), 1175–1186 (2018)
10. Emerick, A.A., Reynolds, A.C.: Ensemble smoother with multiple data assimilation. *Comput. Geosci.* **55**, 3–15 (2012)
11. Evensen, G., Raanes, P.N., Stordal, A.S., Hove, J.: Efficient implementation of an iterative ensemble smoother for big-data assimilation and reservoir history matching. *Front. Appl. Math. Stat.* **5**, 47 (2019)
12. Iglesias, M.A.: Iterative regularization for ensemble data assimilation in reservoir models. *Comput. Geosci.* **19**, 177–212 (2015)
13. Luo, X., Stordal, A., Lorentzen, R., Nævdal, G.: Iterative ensemble smoother as an approximate solution to a regularized

- minimum-average-cost problem: theory and applications. *SPE J.* **20**, 962–982 (2015). <https://doi.org/10.2118/176023-PA>, SPE-176023-PA
14. Ma, X., Bi, L.: A robust adaptive iterative ensemble smoother scheme for practical history matching applications. *Comput. Geosci.* **23**(3), 415–442 (2019)
  15. Stordal, A.S., Elsheikh, A.H.: Iterative ensemble smoothers in the annealed importance sampling framework. *Adv. Water Resour.* **86**, 231–239 (2015)
  16. Evensen, G.: Analysis of iterative ensemble smoothers for solving inverse problems. *Comput. Geosci.* **22**. <https://doi.org/10.1007/s10596-018-9731-y> (2018)
  17. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenom.* **60**(1–4), 259–268 (1992)
  18. Chen, X., Xu, F., Ye, Y.: Lower bound theory of nonzero entries in solutions of  $\ell_2$ - $\ell_p$  minimization. *SIAM J. Sci. Comput.* **32**(5), 2832–2852 (2010)
  19. Li, L., Jiang, S., Huang, Q.: Learning hierarchical semantic description via mixed-norm regularization for image understanding. *IEEE Trans. Multimed.* **14**(5), 1401–1413 (2012)
  20. Park, T., Casella, G.: The Bayesian Lasso. *J. Am. Stat. Assoc.* **103**(482), 681–686 (2008)
  21. Li, Q., Lin, N., et al.: The Bayesian elastic net. *Bayesian Anal.* **5**(1), 151–170 (2010)
  22. Luo, X., Bhakta, T., Jakobsen, M., Nævdal, G.: An ensemble 4D-seismic history-matching framework with sparse representation based on wavelet multiresolution analysis. *SPE J.* **22**, 985–1010 (2017). <https://doi.org/10.2118/180025-PA>, SPE-180025-PA
  23. Luo, X., Bhakta, T., Jakobsen, M., Nævdal, G.: Efficient big data assimilation through sparse representation: A 3D benchmark case study in petroleum engineering. *PLOS ONE* **13**, e0198586 (2018)
  24. Soares, R.V., Luo, X., Evensen, G., Bhakta, T.: 4D seismic history matching: Assessing the use of a dictionary learning based sparse representation method. *J. Pet. Sci. Eng.* **195**, 107763 (2020)
  25. Lorentzen, R., Flornes, K., Nævdal, G.: History matching channelized reservoirs using the ensemble Kalman filter. *SPE J.* **17**, 137–151 (2012)
  26. Canchumuni, S.W.A., Emerick, A.A., Pacheco, M.A.C.: History matching geological facies models based on ensemble smoother and deep generative models. *J. Pet. Sci. Eng.* **177**, 941–958 (2019)
  27. Jafarpour, B.: Wavelet reconstruction of geologic facies from nonlinear dynamic flow measurements. *IEEE Trans. Geosci. Remote Sens.* **49**, 1520–1535 (2011)
  28. Sarma, P., Chen, W.H., et al.: Generalization of the ensemble Kalman filter using kernels for nongaussian random fields. In: *SPE reservoir simulation symposium*. Society of Petroleum Engineers, The Woodlands. SPE-119177-MS (2009)
  29. Strebelle, S.: Conditional simulation of complex geological structures using multiple-point statistics. *Math. Geol.* **34**, 1–21 (2002)
  30. Cover, T.M., Thomas, J.A.: *Elements of information theory*. Wiley (2012)
  31. Luo, X., Bhakta, T.: Automatic and adaptive localization for ensemble-based history matching. *J. Pet. Sci. Eng.* **184**, 106559 (2020)
  32. Magnus, J.R., Neudecker, H.: *Matrix differential calculus with applications in statistics and econometrics*. Wiley (2019)
  33. Simon, D.: *Optimal state estimation: Kalman, H-infinity, and nonlinear approaches*. Wiley-Interscience (2006)

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.