Real-time data assimilation for large-scale systems: the Spectral Kalman Filter

Hojat Ghorbanidehno^{1,*}, Amalia Kokkinaki², Judith Yue Li², Eric Darve^{1,3},
Peter K. Kitanidis^{2,3}

Abstract

The Kalman Filter (KF) is a data assimilation method that has been widely used for estimating spatially varying unknown states evolving in time. Recently, KF methods have shown potential for tracking CO_2 plumes injected in deep geologic formations. Such real-time estimation would serve as an early warning system for leakage incidents. However, for the large number of unknowns of such large systems, KF methods are impractical, because updating the huge state covariance matrix is computationally expensive. Low rank approximation methods have been devised to overcome this problem; these methods assume a low rank of the covariance matrix, which they approximate by smaller matrices. The approximation error is small for smooth functions, but may be larger for more complex physical problems, potentially

*Corresponding author

Email addresses: `hojjatgh@stanford.edu` (Hojat Ghorbanidehno),
`amaliak@stanford.edu` (Amalia Kokkinaki), `yuel@stanford.edu` (Judith Yue Li),
`darve@stanford.edu` (Eric Darve), `peterk@stanford.edu` (Peter K. Kitanidis)

¹Department of Mechanical Engineering, Stanford University, 496 Lomita Mall, Stanford, CA, 94305, USA

²Department of Civil and Environmental Engineering, Stanford University, 473 Via Ortega, Stanford, CA, 94305, USA

³Institute for Computational and Mathematical Engineering, Jen-Hsun Huang Engineering Center, Stanford University, 473 Via Ortega, Stanford, CA, 94305, USA

Preprint submitted to Advances in Water Resources

July 28, 2015

leading to filter divergence and inaccurate state estimates.

We present the Spectral Kalman Filter (SpecKF), a new algorithm that utilizes the exact covariance matrix. Thus it avoids the approximation error, and does so in a computationally efficient way, which is specially important for systems with large numbers of unknowns. The computational speed-up of the SpecKF is achieved by updating cross-covariance matrices instead of the larger covariance matrices. The benefit can be considerable, especially in large systems, because the computational complexity of the SpecKF scales with the number of measurements, as opposed to the effective rank of the covariance matrix in low-rank KFs or the number of ensemble members in ensemble methods. We investigate the accuracy and performance of the SpecKF for a diffusion problem with random perturbations, and for a more complex case of CO_2 injection in a homogeneous two-dimensional domain. Our results show that the SpecKF reduces greatly the computational cost compared to the original KF algorithm, and that it can provide higher accuracy than the Ensemble Kalman Filter (EnKF) with the same or even smaller computational cost. Finally, we discuss some approaches that can be used with the SpecKF for approximating the uncertainty of the final state estimates.

Keywords: Kalman filter, Data assimilation, CO_2 monitoring, Bayesian analysis, Uncertainty quantification, Fast Kalman filter

1. Introduction

Data assimilation methods combine dynamical models of a system with noisy observations to obtain estimates of the state of the system with time.

When these dynamical and observation models are linear, the Kalman filter (KF) algorithm gives the best estimate in a least square sense [1]. For non-linear dynamic models, the KF algorithm is not exact, but it can provide reasonably accurate solutions using some linearization scheme. Several non-linear KF variants exist [2, 3, 4]. The Extended Kalman Filter (EKF) is amongst the most popular in geosciences [5, 6, 7]. Specifically, it has found many applications over the past two decades in hydrogeology [8, 9, 10] and geophysics [11, 12] and has proven a powerful technique for assimilating new measurements to continuously update the estimate of state variables. In such applications, filtering refers to the use of observations for the sequential and real-time improvement of the predictions obtained by imperfect models, where imperfections are due to simplified model formulation and/or unknown model parameters. Subsurface physical processes are particularly well suited for Kalman Filter techniques, because of the unknown spatial heterogeneity of soil properties and its corresponding effect on the states of interest (e.g., the pressure distribution). However, one of the major limitations in such cases is the inherent high dimensionality that arises when such systems are appropriately discretized for modeling. This ‘curse of dimensionality’ becomes further pronounced when the physical problem in question extends over large-scale domains. One such application is the injection and storage of carbon dioxide (CO_2) in deep geologic formations for climate change mitigation. An important requirement for this application is the ability to rapidly estimate the reservoir model state (pressures and saturations), so that potential leakages and other system failures that pose hazards for the operation can be detected and dealt with.

For such systems with a significantly large state space (large number of unknowns), implementation of the original Kalman Filter method is extremely expensive because it requires updating a large covariance matrix every time new measurements are assimilated, which is normally performed at a computational cost of $\mathcal{O}(nm^2)$, where n and m are the number of measurements and the number of unknowns respectively. This quadratic scaling with m can result in prohibitive computational costs even for moderately large systems. This problem has given rise to techniques that focus on reducing the computational cost of the Kalman Filter for large-scale systems. Some of the avenues that have been explored to achieve computational efficiency is the reduction of the cost associated with updating the covariance matrix, and with calculating the Jacobian required for the linearization of non-linear systems.

The Ensemble Kalman Filter (EnKF) is one of the most widely applied KF variants for non-linear cases. It is based on the direct evaluation of the covariance using an ensemble of randomly perturbed realizations, resulting in a low-rank approximation of the state covariance matrix at a computational cost of $\mathcal{O}(nmN)$, where N is the number of ensemble members. This approach simultaneously renders the Jacobian calculation unnecessary [13, 14, 15, 16, 17, 18]. However, while the EnKF is straightforward to implement and highly parallelizable, in some applications it requires a large ensemble size to give accurate estimates, which increases the computational cost. In particular, it has been shown that an ensemble size larger than the number of measurements is needed to avoid filter divergence [19, 20]. As a consequence of this, for problems with large amounts of independent data,

such as 4D seismic (e.g., [21]), covariance localization schemes are necessary to avoid rank deficiency and ensemble collapse, thereby introducing more tuning parameters in the algorithm and increasing the computational cost [22, 23, 24].

Another class of low-rank methods are methods that approximate the covariance using matrix factorization. In these methods, the $m \times m$ covariance matrix is approximated by a low-rank matrix. In essence, these methods project the covariance matrix to a smaller subspace determined by a given set of orthogonal vectors. Then, instead of the original covariance matrix, the filter updates the projection basis at every measurement assimilation. One such method is the Singular Evolutive Extended Kalman filter (SEEK) [28]. In this method, further computational reduction can be achieved by keeping the basis constant during the data assimilation process [29]. Along these lines, [25] proposed the Compressed State Kalman Filter (CSKF), and showed that the choice of the basis can be tailored to the problem of interest to improve estimation accuracy. Application of the CSKF even for highly non-linear systems has shown that significant computational benefits are possible; [27] showed that for a system of 2,025 unknowns, a compression down to only 100 orthogonal bases suffices to provide accurate results, for a strongly non-linear problem of CO_2 monitoring in a heterogeneous domain. However, as discussed in [25], while low-rank methods based on covariance factorization and their variants [28, 11, 29] can effectively reduce the computational requirements for problems where the rank of the state covariance matrix is smaller than the number of unknowns, it may require a large number of orthogonal bases for more complex problems with high frequency components

in the state transition matrix (i.e. the forward model operator). As the number of bases considered increases, the computational benefits of using such low rank methods is less pronounced.

Reducing the computational cost of the EKF is also possible by utilizing fast linear algebra methods for accelerating the matrix-vector multiplications involved in the KF recurrences. Such techniques can be complementary to the EKF variants mentioned above. Fast matrix-vector multiplication methods are indispensable for large systems, as even with modern computers, multiplications between large dense matrices in excess of $\approx 100,000$ elements are infeasible, both in terms of storage and computational cost. A suite of linear algebra techniques are available in the literature, such as Fast Fourier Transform-based approaches for regular grids [30], and Fast Multipole Methods (FMM) [31, 32, 33, 34] applicable also to irregular grids. Such methods can be used to perform the computations required to update the covariance matrices in the Kalman Filter with a significantly smaller computational cost.

Finally, another approach for improving the computational efficiency of the EKF algorithm was recently introduced by [26], who used a modified EKF recurrence to update the small cross-covariance matrices as opposed to the square, large covariance matrices. This approach provides a remarkable speed up in the computations without introducing error by covariance approximations. They combined this approach with fast linear algebra using the FMM method to formulate a fast KF algorithm termed HiKF. However, the HiKF method is only applicable for a random walk forecast model, where the forward matrix is approximated by the identity matrix; while such a simple model may provide reasonable accuracy for near continuous data

assimilation, it is not applicable for the more typical case where the state to be estimated may undergo significant changes between data assimilation steps.

In this paper, we extend the approach of [26] to the general case of non-linear dynamic systems and propose a new Kalman Filter algorithm, which we term Spectral Kalman Filter. Similar to the HiKF, the SpecKF updates cross-covariances instead of covariance matrices. The proposed method is applicable for arbitrary forward models by means of a forward model approximation. This approximation brings about a small and controllable approximation error. The computational cost of estimation with our method is reduced from $\mathcal{O}(nm^2)$ to $\mathcal{O}(n^2m)$, which can be a dramatic decrease, especially for large systems, where $n \ll m$. The remaining matrix-vector multiplications involving the error covariance matrix are accelerated by a low-rank approximation of the *error* covariance matrix using eigenvalue decomposition method. In doing this, only the largest eigenvalues of the *error* covariance matrix are used, while the rest are neglected. In that sense, the performance of our method is related to the spectrum of the *error* covariance matrix. Overall, the performance of our method depends on the spectral properties of the error covariance, and the forward model approximation. The computational cost of the new method scales linearly with the number of unknowns, while the computational cost of full KF method scales quadratically with number of unknowns. Finally, for the estimation of the Jacobian, the SpecKF utilizes a matrix-free approach, which uses the forward model as a "black box" and is fully parallelizable. As a result, our method can handle a large number of unknowns in a computationally efficient way, both

1
2
3
4
5
6
7
8
9 in terms of storage and computational time.

10
11 The paper is organized as follows. We start by a brief presentation of
12
13 the Kalman Filter, followed by the algorithm and implementation details
14
15 of the newly proposed SpecKF. Then, the performance and accuracy of the
16
17 SpecKF is compared with the EnKF. We use two benchmark applications
18
19 for this evaluation: in the first case, a linear dynamic system described by a
20
21 perturbed diffusion equation is investigated to compare the performance of
22
23 the SpecKF with the EnKF. The two algorithms, both being approximations
24
25 of the original KF algorithm, are compared to the reference solution of the
26
27 KF in terms of accuracy and computational performance. In the second case,
28
29 we evaluate the SpecKF for a non-linear data assimilation problem, where
30
31 the algorithm is used to monitor the development of CO_2 injection with time
32
33 in a homogeneous, two-dimensional domain. We compare the accuracy and
34
35 computational efficiency of our method to that of the EnKF, showing that
36
37 even for non-linear cases, the SpecKF can provide better accuracy for the
38
39 same computational cost as the EnKF. Finally, we discuss some approaches
40
41 that can be used with the SpecKF for approximating the uncertainty of the
42
43 final state estimates.
44

45 **2. Methods**

46 *2.1. Kalman Filter*

47
48 The original Kalman Filter algorithm was developed to estimate the state
49
50 of a linear time-discretized system based on its previous state and measure-
51
52 ments [1]. For a linear dynamical system, the filter is optimal in a least square
53
54 sense, resulting in the best possible estimate given the available measure-
55
56
57
58
59
60
61
62
63
64
65

ments. The Kalman filtering process has two steps: (i) the *prediction/forecast* of the state at time t_{k+1} using the forward model and the current estimate at time t_k , and (ii) the *update/analysis* of the predicted state at time t_{k+1} based on newly available measurements. The prediction step predicts the state of the system at the next time step using *only* information from the current time step. The estimate obtained from the prediction step is termed as the *a priori* state estimate, and usually includes errors due to forward model limitations (e.g., unknown parameters or initial conditions). The update step combines this *a priori* prediction with the measurements obtained at the next time step, to refine the estimate of the state of the system at that time. The estimate obtained from the update step is termed as the *a posteriori* state estimate. The *a posteriori* state estimate can be thought of as a weighted average of the *a priori* state estimate and the estimate of the system obtained from the measurements.

Consider a state estimate vector $s_k \in R^m$ and a vector of noisy measurements $z_k \in R^n$ at time t_k . The state of a linear dynamic system model can be represented as:

$$s_{k+1} = F_{k+1}s_k + v_{k+1} \quad (1)$$

where $F_k \in R^{m \times m}$ is the state transition model at time k , and $v_k \sim N(0, Q_k)$ represents the model error or noise. Assuming that s_k is Gaussian with a mean equal to the best estimate $\hat{s}_k$ and error covariance matrix P_k , an equation for the propagation of the error covariance matrix can be derived as:

$$P_{k+1} = F_{k+1}P_kF_{k+1}^T + Q_{k+1} \quad (2)$$

The measurement vector z_k is related to the true state vector through an

176 observation model H:

$$z_{k+1} = H_{k+1}s_{k+1} + w_{k+1} \quad (3)$$

177 where $H \in R^{n \times m}$ is the observation model matrix and $w_k \sim N(0, R_k)$ is a
 178 vector containing unbiased measurement noise. Then the Kalman Filtering
 179 process is given in the following recurrence (f: forecast, a: analysis):

181 Prediction/Forecast:

$$\hat{s}_{k+1}^f = F_{k+1}\hat{s}_k^a \quad (4a)$$

$$P_{k+1}^f = F_{k+1}P_k^a F_{k+1}^T + Q_{k+1} \quad (4b)$$

183 Update/Analysis:

$$K_{k+1} = P_{k+1}H_{k+1}^T \left(H_{k+1}P_{k+1}^f H_{k+1}^T + R_{k+1} \right)^{-1} \quad (4c)$$

$$\hat{s}_{k+1}^a = \hat{s}_{k+1}^f + K_{k+1} \left(y_{k+1} - H_{k+1}\hat{s}_{k+1}^f \right) \quad (4d)$$

$$P_{k+1}^a = P_{k+1}^f - K_{k+1}H_{k+1}P_{k+1}^f \quad (4e)$$

186 The Kalman Gain matrix K (Eq. (4c)) contains the influence of measure-
 187 ments on the different elements of $\hat{s}_k$ and determines how much the innovation
 188 term (or residual), given by $y_{k+1} - H_{k+1}\hat{s}_{k+1}^f$, is weighted. Then, the elements
 189 in the updated or analyzed state vector s_{k+1}^a are given by a linear combina-
 190 tion of the predicted state vector (s_{k+1}^f) and the innovation term from the
 191 measurements (Eq. (4d)).

For non-linear problems, a non-linear function $f(\cdot)$ may represent the
 relationship, or sensitivity of the state at time $k + 1$ to the state at time

k . Similarly, the measurements may be related to the state with a non-linear function $h(\cdot)$, such that: $z = h(s) + w$. In such cases, the Extended Kalman Filter is used; the F and H matrices in Eq. (4d) and Eq. (4a) are replaced by the non-linear functions $f(\cdot)$ and $h(\cdot)$, i.e., $Fs_k^a \rightarrow f(s_k^a)$ and $HS_{k+1}^f \rightarrow h(s_{k+1}^f)$. The matrices F and H in all other equations are the Jacobian matrices of the forward model and measurement model respectively at each time step:

$$F_k = \left. \frac{\partial f}{\partial s} \right|_k, \quad H_k = \left. \frac{\partial h}{\partial z} \right|_k$$

The Jacobian matrices F and H may be obtained by analytical differentiation in simple cases. When analytical differentiation is not possible (as in most complex non-linear dynamical models), numerical differentiation is used, which can be computationally demanding. Adjoint methods can be used to reduce this computational cost, but they require modification of the forward model code and are not applicable for arbitrary boundary conditions. Alternatively, numerical differentiation can be performed in a computationally efficient way using a matrix-free approach (see [35]), which is described in the following section, along with the algorithm and implementation details of the Spectral Kalman Filter.

2.2. Spectral Kalman Filter

The implementation of the EKF algorithm for large-scale systems becomes infeasible, because of the high storage cost and computational cost of the matrix operations outlined in Eq. (4b) to Eq. (4e), involving several multiplications of $m \times m$ matrices when updating the covariance matrix P . In low-rank methods, the $m \times m$ matrix P is approximated by multiplication

of much smaller matrices, which serves as the low-rank approximation of the covariance. This approximation can bring about errors that, in some cases, may lead to inaccurate estimates, rank deficiency and filter collapse. To avoid the consequences of approximating the covariance P , we propose an alternative algorithm that implicitly uses the full matrix P , while it reduces the computational cost of the Kalman Filter by modifying the recurrence matrix operations such that only the $m \times n$ cross-covariance matrices (such as PH^T) are stored and updated at every time step. This modification takes advantage of the Kalman Filter recurrence to calculate products of big matrices like PH^T without explicitly having to construct and multiply the individual matrices. This is achieved by multiplying Eq. (4b) and Eq. (4e) by appropriate Jacobians to obtain (hereafter the subscripts k are dropped for F , H , and Q for readability):

$$P_{k+1}^f H^T = F P_k^a F^T H^T + Q H^T \quad (5)$$

$$P_{k+1}^a H^T = P_{k+1}^f H^T - K_{k+1} H P_{k+1}^f H^T \quad (6)$$

However, after the above modifications, Eq. (5) still involves the product $P_k^a F^T H^T$, which is computationally intractable for large m . In order to overcome this problem, we store and update the term $P^a F^T H^T$ by Expanding this term and using Eq. (4b) and Eq. (4e) as explained below:

$$\begin{aligned} P_{k+1}^a F^T H^T &= (I - K_{k+1}) P_{k+1}^f F^T H^T \\ &= (I - K_{k+1} H) (F P_k^a F^T H^T + Q F^T H^T) \end{aligned} \quad (7)$$

It becomes apparent that, while Eq. (5) and Eq. (6) can be used to update the cross-covariances using the recurrence rather than direct matrix-vector

1
 2
 3
 4
 5
 6
 7
 8
 9 multiplications, the term $P_{k+1}^a F^T H^T$ invalidates this benefit as it involves
 10 terms that are equally expensive to compute (i.e. $F P_k^a F^T F^T H^T$). Therefore,
 11 we rewrite the $P_k^a F^T F^T H^T$ term as below (see Appendix A for derivation):
 12
 13
 14

$$\begin{aligned}
 P_k^a F^T F^T H^T &= P_k^a ((F - I) + I)^T H^T \\
 &= -P_k^a H^T + 2P_k^a F^T H^T + P_k^a ((F - I)^T H^T) \quad (8)
 \end{aligned}$$

15
 16
 17
 18
 19
 20
 21 To simplify the equations and construct our fast method, we will need to
 22
 23 assume that $F - I$ is small. Recall that F is the Jacobian of the forward
 24
 25 model f with respect to the unknowns s . F is close to I when dt is small
 26
 27 and more generally:⁴
 28

$$F - I \approx \mathcal{O}(dt) \quad (9)$$

29
 30
 31 Using the above, we obtain the following:
 32

$$P_k^a F^T F^T H^T \approx -P_k^a H^T + 2P_k^a F^T H^T + \mathcal{O}((dt)^2) \quad (10)$$

33
 34
 35
 36
 37 where $(F - I)^2 \sim \mathcal{O}((dt)^2)$. As a result, for a small dt , the error intro-
 38
 39 duced by neglecting the last term of Eq. (10) is expected to be small. Most
 40
 41 importantly, having an explicit formula for this error, we can control and
 42
 43 reduce it by increasing the frequency of the prediction step of the Kalman
 44
 45 filter, or by adding more recurrence terms to the SpecKF. We note that at
 46
 47 each step, we are making an $\mathcal{O}((dt)^2)$ error. This means that there is a global
 48
 49 error for the time discretization of $\mathcal{O}(dt)$ (the method is first order). In our
 50
 51 benchmark examples, this accuracy was sufficient. If a higher-order scheme
 52

53
 54 ⁴This can be proved using a Taylor expansion. The only assumption that is required
 55 is the existence of the derivative $\partial F / \partial t$.
 56

is required, the derivation for including more recurrence terms is given in the appendix. However, since this approximation is used only once within the algorithm (i.e., the exact operator F is used in all other calculations except in the expansion of Eq. (10)), depending on the non-linearity of the problem, the error introduced by this forward model approximation will likely not affect the results of the filter significantly.

In this paper, dt is assumed to be the time interval between data assimilations. However, more generally, dt can be set equal to the time-step used in the forward model, and can be chosen independently from the time interval between data assimilations. In such an approach, we use a sub time-stepping strategy, in which we perform several prediction/forecast steps with no update/analysis. When we reach a point in time where observations are provided, the normal Kalman Filter step is applied. This approach allows choosing a dt that is small enough to guarantee a small error in our method. This choice is now independent of the data assimilation time. This demonstrates that requiring F to be close to I is an assumption that is relatively easy to satisfy in practice, and is not a real restriction.

Using Eq.(10) and defining the new matrices $T_{k+1} = P_{k+1}^f H^T$, $P_{k+1}^H = P_{k+1}^a H^T$, and $P_{k+1}^F = P_{k+1}^a F^T H^T$, the SpecKF algorithm can be summarized

by the following recurrence:

$$\hat{s}_{k+1}^f = f(\hat{s}_k^a) \quad (11a)$$

$$T_{k+1} = FP_k^F + QH^T \quad (11b)$$

$$K_{k+1} = T_{k+1}(HT_{k+1} + R)^{-1} \quad (11c)$$

$$\hat{s}_{k+1}^a = \hat{s}_{k+1}^f + K_{k+1}(z_{k+1} - h(\hat{s}_{k+1}^f)) \quad (11d)$$

$$P_{k+1}^F = (I - K_{k+1}H)(-FP_k^H + 2FP_k^F + QF^TH^T) + \mathcal{O}((dt)^2) \quad (11e)$$

$$P_{k+1}^H = (I - K_{k+1}H)T_{k+1} \quad (11f)$$

Inspection of Eq. (11a) to Eq. (11e) indicates that explicit multiplications between the covariance P and other matrices or vectors are avoided entirely, since these products (the cross-covariances) are updated through the recurrence. Thus, even though the entire covariance matrix P_k is never stored or computed, it is still considered in its exact form in the computations. The forward model error introduced in Eq. (9) is only affecting the solution once in the update of the P^F matrix.

The only large matrix-vector multiplications are the terms QH^T and QF^TH^T . These operations can be performed using a low-rank approximation of the Q matrix. If the Q matrix (i.e. the error covariance) is smooth and its spectrum decays fast, a low rank approximation can provide very good accuracy by keeping only the first few terms of the spectrum, and thereby providing significant computational benefits. However, even in the case of a non-smooth Q , using a low-rank approximation, will still be appropriate if we are mainly interested in recovering the dominant features of the unknown field. In addition, the error covariance Q is assumed to be constant in time,

therefore the low-rank approximation is performed only once at the beginning. It should be noted that this low-rank approximation is not the same as the low-rank approximation of the P matrix, which is typically done by low-rank Kalman filter methods, such as EnKF. Low rank approximation of the P matrix is less justifiable because the P matrix is changing with time and it may not be as smooth as Q .

One important aspect of the SpecKF algorithm is the calculation of the F and H operators, which represent the measurement and dynamic model Jacobian matrices. As previously mentioned, obtaining the Jacobian for non-linear problems is non-trivial and computationally intensive. One common way is to numerically approximate the derivatives:

$$F = \frac{df}{ds} \approx \frac{f(s + \delta) - f(s)}{\delta} \quad (12)$$

$$H = \frac{dh}{ds} \approx \frac{h(s + \delta) - h(s)}{\delta} \quad (13)$$

Upon calculation of the Jacobian in this manner (which requires $m+1$ forward model runs), one would need to perform matrix-matrix products which are also expensive for large m . To perform these calculations more efficiently, a matrix-free approach has been used [35]. The matrix-free approach uses numerical differentiation, but instead of separately calculating the Jacobian matrix and then multiplying it with another matrix or vector x , it calculates directly the product of the Jacobian and x . Consider we want to compute Fx where $F = \frac{\partial f}{\partial s}$ at $s = s_0$, and x is a vector. Then, Fx and Hx are computed directly by:

$$Fx \simeq \frac{\|x\|}{\delta\|s_0\|} \left[f\left(s_0 + \frac{\|s_0\|}{\|x\|}\delta x\right) - f(s_0) \right] \quad (14)$$

$$Hx \simeq \frac{\|x\|}{\delta\|s_0\|} \left[h\left(s_0 + \frac{\|s_0\|}{\|x\|}\delta x\right) - h(s_0) \right] \quad (15)$$

271 where the ratio of Euclidean norms $\frac{\|x\|}{\|s_0\|}$ is used as a normalization factor
 272 and δ is a small dimensionless parameter. Calculating Fx in this manner has
 273 several advantages: it uses the forward simulation as a black-box, such that
 274 no modifications in the code are required. Furthermore, it only requires one
 275 forward run for each vector x , such that for example the computation of FP^F
 276 where F is a $m \times m$ matrix and P^F is a $m \times n$ matrix, has a computational cost
 277 equal to $n + 1$ calls to forward solver (the number of measurements plus one).
 278 Finally, these computations can be performed independently and in parallel
 279 to further increase computational efficiency. The choice of the parameter
 280 δ is important for the performance of the matrix-free approach. A large
 281 value for δ will cause loss of accuracy in the derivative and computational
 282 instabilities. However, very small values of δ lead to machine roundoff errors,
 283 and the smallest value depends on the floating point number precision (see,
 284 e.g., [35]). The δ value for the applications presented here, along with other
 285 parameters of the SpecKF used, can be found in Table 1. The computational
 286 cost of the SpecKF for non linear problems, measured in runs of the forward
 287 model, is shown in Table 2. For comparison, the computational cost of EnKF
 288 is also given.

289 While the strength of the SpecKF lies in avoiding the computationally
 290 intensive propagation of the error covariance matrix P by considering the
 291 cross-covariance PH^T instead, it comes at the cost of not having access to
 292 the full covariance matrix of the estimated state. The diagonal elements of
 293 this covariance matrix P represent the uncertainty of estimation and might
 294 be desirable to have for uncertainty quantification. In such cases, there are
 295 two approaches that can be followed to obtain the desirable uncertainties. In

Table 1: Kalman Filter parameters for the CO_2 benchmark application

Parameter	Value	Units
<i>Spectral Kalman Filter</i>		
σ_P	$4 \cdot 10^4$	Pa
σ_S	$2 \cdot 10^{-2}$	(-)
σ_P	$2 \cdot 10^5$	
σ_S	$2 \cdot 10^{-2}$	
L	450	m
δ^a	0.001	

^a δ used only in the SpecKF

the case that we are interested in the uncertainty of the state estimate at one or more specific locations (e.g., near a fault or a well), we can add a control point at this location. This is done in our implementation by simply adding a new measurement point with an infinitely large uncertainty. While inclusion of the control points will not affect the estimate during the recursion, it will provide an additional non-zero element in the H matrix, through which the corresponding diagonal element of P can be obtained from PH^T .

Alternatively, one might be willing to sacrifice some accuracy in the uncertainty estimation to obtain a more complete picture of the uncertainty by having the full P matrix. This can be done if we assume that the main features of the P matrix can be effectively represented by a low-rank matrix. In this case, the known matrix HPH^T can be used to obtain an approximation of P , after projecting the latter on a smaller subspace using with a pre-determined basis (Section C). It should be noted, that this approxima-

Table 2: Comparison of computational cost of the SpecKF and the EnKF for non linear problems measured in terms of number of forward model calls. N : ensemble members, n : number of measurements, K : effective rank of the Q matrix.

Operation	Cost
SpecKF	
$s_{k+1}^f = f_{k+1}(s_k^a)$	1
$T_{k+1} = F_{k+1}P_k^F + Q_{k+1}H^T$	$n + K$
$K_{k+1} = T_{k+1}(H_{k+1}T_{k+1} + R_{k+1})^{-1}$	n
$s_{k+1}^a = s_{k+1}^f + K_{k+1}\left(y_{k+1} - h_{k+1}(s_{k+1}^f)\right)$	1
$P_{k+1}^F = (I - K_{k+1}H)(2F_kP_k^F - FP_k^H + QF^TH^T)$	$3n + 2K$
$P_{k+1}^H = T_{k+1} - K_{k+1}H_{k+1}T_{k+1}$	-
Total	$5n + 3K$
EnKF	
$s_{k+1,j}^f = f_{k+1}(s_{k,j}^a) + w_{k+1,j} \quad j = 1, \dots, N$	N
$D_{k+1} = [s_{k+1,1}^f - \bar{s}_{k+1}^f, \dots, s_{k+1,N}^f - \bar{s}_{k+1}^f]$	-
$P_{k+1}^f = \frac{1}{N-1}D_{k+1}D_{k+1}^T$	-
$K_{k+1} = P_{k+1}^f H^T (HP_{k+1}^f H^T + R)^{-1}$	$N + n$
$s_{k+1,j}^a = s_{k+1,j}^f + K_{k+1}(y_{k+1} - h_{k+1}(s_{k+1,j}^f) + v_{k+1,j})$	N
$s_{k+1}^a = \frac{1}{N} \sum_{j=1}^N s_{k+1,j}^a$	-
Total	$3N + n$

tion of the posterior covariance is a post-processing step, and does not affect the estimation obtained from the SpecKF.

2.3. Applications

2.3.1. Validation problem

For a linear problem, the KF provides the best estimate in the least-square sense for the data assimilation problem. Therefore, to validate the SpecKF implementation, we compare it to the results obtained from the original KF algorithm; the difference between the two indicates how significant is the combined error of the forward model approximation and the low-rank approximation of Q , which are the two sources of error in the SpecKF algorithm. The benchmark problem we use is the linear diffusion equation in a two-dimensional unit domain, $[0, 1] \times [0, 1]$, where the state is contaminated with random noise:

$$\frac{\partial \phi}{\partial t} = D \nabla^2 \phi + \text{noise} \quad (16)$$

In Eq. (16), D represents the diffusion coefficient which here is assumed to be spatially and temporally constant. When discretized, Eq. (16) may be written in the form of Eq.(1). Discretizing Eq. (16) using explicit Euler in time and central difference in space, we obtain:

$$\begin{aligned} \phi_{i,j}^{(k+1)} = \phi_{i,j}^{(k)} + \frac{dt}{(dx)^2} \left(\phi_{i+1,j}^{(k)} - 2\phi_{i,j}^{(k)} + \phi_{i-1,j}^{(k)} \right) + \\ \frac{dt}{(dy)^2} \left(\phi_{i,j+1}^{(k)} - 2\phi_{i,j}^{(k)} + \phi_{i,j-1}^{(k)} \right) + \sqrt{dt} w_{i,j}^{(k)} \end{aligned} \quad (17)$$

where $\sqrt{dt}w$ is noise that represents various small and erratic sources and sinks over the domain with zero mean and a covariance matrix of dtQ . In

this problem, Q is selected from the Gaussian family of covariance func-
 tions, where $Q_{i,j} = 10 \exp(-(\frac{x_i - x_j}{L})^2)$. Eq. (17) is solved for a 51×51
 grid resulting in $m = 2401$ (excluding the boundaries), with a discretization
 $dx = dy = 0.02$, with Dirichlet boundary conditions $\phi = 1$ at all boundaries,
 and initial $\phi = 0$ throughout the domain except at boundaries. For the time
 discretization, $dt = 10^{-5}$ is chosen, to satisfy the numerical stability require-
 ment $dt \leq 4 \times 10^{-4}$. There are 20 observation points uniformly distributed
 in the unit domain, with observation error covariance $I_{20 \times 20}$. Figure 1 shows
 the distribution of ϕ at four different time steps, indicating the distortion of
 an otherwise smooth field due to the noise added.

Figure 1: Perturbed true solution used for validation of the SpecKF at a) time step 400,
 b) time step 800, c) time step 1000, and d) time step 1500

In addition to the comparison to the KF results for validation of the SpecKF, the diffusion problem is used to conduct error analysis and evaluate the computational efficiency of the SpecKF compared to both the KF and the EnKF.

2.3.2. Physical problem: CO_2 monitoring

The second benchmark problem used to evaluate the SpecKF is for the strongly non-linear physical problem of multiphase transport in the subsurface. Specifically, the objective is to use the SpecKF to estimate the pressure and saturation distribution with time in a domain where supercritical CO_2 is being injected in a synthetic two-dimensional CO_2 injection experiment.

The forward model used to simulate the injection of CO_2 in the subsurface is TOUGH2, a multi-dimensional, multi-phase, multi-component non-isothermal numerical simulator for flow and transport in porous media [36, 37]. The TOUGH2 family of codes along with its various equation-of-state modules, has been validated, tested and widely used in the literature for numerous applications [38]. Specifically for injection of CO_2 in the subsurface, TOUGH2 is used in combination with the ECO2N module [39, 40], which is designed to simulate two-phase (brine- CO_2) systems for conditions of pressure and temperature under which CO_2 is above the critical point, as is typical for injection of CO_2 in deep geologic formations. The ECO2N module has been used in many carbon sequestration modeling studies to evaluate the potential for carbon storage in deep saline aquifers [41, 42], to simulate the evolution of injected CO_2 plumes [43, 44], and the physical processes that affect this evolution [45, 46, 47, 48], to model leakage of CO_2 in shallow groundwater [49, 50], as well as to validate newly developed numerical

models for CO_2 transport [51, 52].

In the present work, we use the parallel (MP) version of TOUGH2-ECO2N for the forward simulations in order to increase the computational efficiency of the inversion. As noted previously, the cost of the forward simulation is more significant compared to the cost of the SpecKF algorithm, and therefore parallelization of the forward problem is highly beneficial. The synthetic CO_2 injection scenario used is illustrated in Figure 2. A homogeneous two-dimensional reservoir is assumed with dimensions $450\text{ m} \times 450\text{ m}$ and a discretization of 10 m in both directions. Four injection wells are considered near the center of the domain (illustrated by the blue circles in Figure 2). CO_2 is injected continuously for 255 days at a constant rate of 1.15 kg/sec per well, for a total injection of $\sim 400\text{ tonnes/day}$. Two production wells are used at opposite corners (noted by red dots in Figure 2). The production wells are at a constant pressure of 200 bar . The reservoir is initially at a constant pressure of $207 \times 10^5\text{ Pa}$ with a uniform saturation of CO_2 , $S_{CO_2} = 0.01$.

The hypothetical reservoir is isotropic with an intrinsic permeability of $k = 3 \times 10^{-12}\text{ m}^2$ and a porosity of 20%. The characteristic curves for capillary pressure and relative permeability are modeled using the van Genuchten function and the van Genuchten-Mualem model respectively [53, 54]. Salt precipitation and other geochemical reactions are assumed to be negligible in the time frame of the experiments considered here and are therefore not modeled. Interphase partitioning between water and CO_2 is assumed to be at equilibrium. Temperature is considered to be constant at 41° C during the CO_2 injection test to increase computational efficiency, consistent with

Figure 2: CO_2 saturation distribution for synthetic experiment after 15 days. Red dots indicate production wells and blue dots indicate injection wells.

previous studies [44].

Initially the forward simulation is run for 255 days to provide the true data for pressure and CO_2 saturation, which are the two state variables we are interested in tracking with the Kalman Filter. From this data, 25 saturations and 9 pressures ($n = 34$) are collected every 15 days. These measurements are then contaminated with Gaussian error (Table 1). Using these measurements, the Kalman Filter algorithms are used to estimate the pressure and saturation for the whole domain, resulting in $2 \times 2025 = 4050$ unknowns ($m = 4050$). Since the reservoir was initially at a constant pressure with uniform saturation, we start the assimilation of data on day 30 of the true experiment to have a more realistic initial condition for our experiment, i.e. the Kalman filter starts estimating system state from day 30. The state of the system at this time would be unknown in practice. For this reason, we initialize the Kalman Filter from a wrong initial condition corresponding to

399 an approximate knowledge of the location of the CO_2 based on the location
 400 of the injection wells. The objective of the Kalman Filter is to predict the
 401 states at subsequent times, based on these erroneous initial conditions, and
 402 the noisy measurements collected. It should be noted that no transformation
 403 or scaling of the saturations was performed in our algorithm. Transformation
 404 of variables or scaling is typically necessary because the Kalman Filter may
 405 give unphysical saturation values below zero or above one [e.g., 55]. Here,
 406 such unphysical predictions were not frequent, and in the few grid points
 407 (less than 10 points in the whole domain at each step) where this happened,
 408 we substituted only the non-physical saturation values with those obtained
 409 from the forward prediction.

For this benchmark application, the results of the SpecKF cannot be
 directly compared with the Extended Kalman Filter because of the compu-
 tational cost of the full EKF. However, we compare the results with results
 obtained by the Ensemble Kalman Filter. The implementation of the EnKF
 is performed based on the method introduced in [14] that is described in the
 appendix. In order to have a valid comparison, parameters of the SpecKF
 and the EnKF are selected to be the same and can be found in Table 1,
 and Eqs. (18), (19), and (20). The SpecKF and the EnKF are compared in
 terms of the accuracy, robustness and computational cost. First, we explored
 the case where the computational cost is the same. In this case, there is 34
 measurements and an effective rank of Q equal to 13 is used, such that the
 SpecKF requires 209 forward model runs. Therefore, the EnKF with an en-
 semble size $N = 58$ has the same computational cost as the SpecKF. Then,
 we investigate the effect of increasing number of measurements on these two

methods.

$$Q = \begin{bmatrix} \sigma_P^2 Q & \sigma_P \sigma_S Q \\ \sigma_P \sigma_S Q & \sigma_S^2 Q \end{bmatrix} \quad (18)$$

$$R = \begin{bmatrix} \sigma_P^2 I_{9 \times 9} & 0 \\ 0 & \sigma_S^2 I_{25 \times 25} \end{bmatrix} \quad (19)$$

$$Q_{i,j} = \exp\left(-\left(\frac{x_i - x_j}{L}\right)^2\right) \quad (20)$$

3. Results and Discussion

3.1. Validation problem

The first benchmark application we investigate is the linear diffusion problem. Figure 3 shows the estimated ϕ obtained with the SpecKF at time step 1000. Comparison to Figure 1c indicates that the SpecKF provides high accuracy results.

Figure 3: SpecKF estimation at time step 1000.

Comparison to the KF results (Figure 4) shows that the difference between the estimations by the SpecKF and the KF is negligible, with the absolute difference $\frac{\|\phi_{SpecKF} - \phi_{KF}\|}{\|\phi_{KF}\|}$ in the order of 10^{-3} , where the true state varies from -1 to 1.5 . This shows that the result of SpecKF agrees with those of KF and the error of the forward model approximation in this case is small. As noted previously, in this case the time step being used is rather small, representing an optimistic case for data assimilation. Nevertheless, the good agreement of SpecKF with KF under these conditions validates the SpecKF implementation. As shown in Figure 4, the overall error added is maintained at low levels, since at all times the time step taken is small (for numerical stability purposes as outlined previously), and since the error of the forward model approximation is second order in time. The contribution of the low-rank approximation of the Q matrix is also included in this error. It should be mentioned that in this problem the SpecKF provides estimation with the same accuracy as the KF method, while the computational cost of SpecKF is much smaller than the computational cost of KF method, specially for the cases with large number of unknowns (Figure 5).

The accuracy of the estimates obtained by Kalman Filter algorithms strongly depends on whether the propagation of the covariance matrix is performed accurately. As the SpecKF involves an approximation error related to the propagation of the cross-covariance (Eq. (11e)), it is important to demonstrate that the covariance terms estimated by the SpecKF also agree with the optimal solution of the KF. In the SpecKF, the P matrix is not propagated; however, the cross covariance matrix PH^T is available, and can be used to obtain the diagonal elements of P that correspond to the measure-

Figure 4: a) Relative difference between the SpecKF and the KF estimates with time, and
b) Absolute difference between the SpecKF and the KF estimations at time step 1000

ment locations (since each row of H matrix has only one non-zero element). This was done in Figure 6. Figure 6a shows the ratio of the SpecKF estimate to the corresponding KF estimate of the error variance for each of the 20 measurement points at a specific time step. Figure 6b shows the ratio of the SpecKF estimate to the KF estimate of the error variance averaged over the 20 measurement points for each data assimilation step. The ratio starts at one as the initial P matrix was the same for both methods. As expected, after a few steps the ratio increases, because an $\mathcal{O}((dt)^2)$ error is affecting

Figure 5: Comparison of time taken by the full KF and the SpecKF methods for different number of unknowns

the SpecKF estimate. However, even though the ratio increases, it remains close to one and the rate of increase slows down significantly as the number of steps increases. This shows that the variance of error in both methods is consistent and that the SpecKF variance is correctly propagated in time.

According to the SpecKF algorithm, this method propagates the cross-covariance PH^T matrix instead of propagation of the error covariance matrix P to reduce the computational cost. Therefore, it is not possible to directly obtain diagonal elements of the covariance matrix P that represent the uncertainty of state estimation. However, the desirable uncertainties still can be achieved by using the two methods explained in the previous section. As described before, when the uncertainty of the state estimate at certain points is important, we can add a control point at these locations and obtain desirable uncertainties of estimate accurately as shown in Figure 7. Also, an approximation of the full posterior covariance matrix can be restored in a post-processing step (Appendix C). Figure 8 shows the approximate pos-

Figure 6: a) The ratio of error variance from SpecKF over KF method at 20 observation points at time step 1,000; b) The ratio of average error variance from SpecKF over KF method at the observation points as function of time.

terior covariance obtained using the PHT^T matrix of the SpecKF, and the projection method described in Appendix C, as well as the exact posterior covariance obtained with the full Kalman Filter.

To demonstrate that the error of the SpecKF behaves as expected, and is $\mathcal{O}(dt^2)$ with respect to KF method based on Eq. (9), we compare the results of SpecKF with KF after the first time step using a range of different time steps. The difference should increase quadratically (see reference line) as the time step increases. This is shown in Figure 9.

Finally, we compare the results of the SpecKF for the diffusion problem with EnKF, an alternative fast Kalman Filter algorithm, in terms of accuracy

Figure 7: Comparison of the uncertainty estimation in KF and SpecKF methods at five control points

Figure 8: (a) Covariance matrix obtained from the full KF method; (b) Approximate covariance matrix obtained using the PH^T matrix of SpecKF

Figure 9: Difference between the KF estimation and the SpecKF estimation versus dt , the time between data assimilation steps.

of estimation. The results of each method are compared to the original KF algorithm. We compare the SpecKF results to those obtained by the EnKF, using the same 20 noisy measurements and the same initial condition. Three different ensemble sizes were evaluated, all larger than the number of measurements. As shown in Figure 10, increasing the ensemble size reduced the relative error of the EnKF. However, in all three cases, the EnKF estimations are less accurate than the SpecKF estimation. Since the ensemble sizes were greater than the number of measurements, the computational cost of the EnKF in all three cases was also greater than that of the SpecKF.

3.2. Application: CO_2 monitoring

The second benchmark application evaluates the performance of the SpecKF for the non-linear problem of CO_2 transport in a two dimensional homoge-

Figure 10: Comparison between the relative error of the SpecKF estimation and the EnKF estimation (with 3 different ensemble sizes) with respect to the KF estimation

neous domain. The accuracy of the SpecKF is compared with that of the EnKF. Both filters are initiated with the same erroneous initial conditions and the same set of noisy measurements is assimilated every 15 days. Results are compared to the true state at different times.

Figure 11 compares the true state with the estimates obtained by the SpecKF and the EnKF with same computational cost. The SpecKF after the first data assimilation step does not give an accurate estimate. However, as more measurements are assimilated the estimation improves drastically such that the SpecKF estimate at 120 days greatly resembles the true state. In this figure, there are some artifacts close to the injection wells (the SpecKF estimate at 120 days) because occasionally the forward model produces unrealistic data around these points. Therefore we changed these values to zero, and the values in the figure are obtained as a weighted average of the 0 value and the value inferred from the observations. The relative error of the SpecKF and the EnKF for saturation estimation is compared in Figure 12, which shows the SpecKF provides the same accuracy as the EnKF for

Figure 11: Comparison of true saturation with the SpecKF and the EnKF estimations. First row) The true saturation, second row) The SpecKF estimation for saturation and third row) The EnKF estimation for saturation with the same computational cost

the same computational cost. In this figure the results are compared after day 30, because the assimilation of data starts on day 30. Also, when the noise in measurements is small compared to the noise in the forward model, the accuracy of the EnKF decreases as shown in Figure 13. In contrast, the accuracy of SpecKF increases which shows that the SpecKF provides higher accuracy than the EnKF with the same computational cost. From this it can be concluded that the computational cost of EnKF is larger than the computational cost of the SpecKF when the same accuracy is desirable and the measurement error is small. Also, comparison of Figure 13 with Figure

Figure 12: Relative error of the EnKF saturation estimation and the SpecKF saturation estimation compared to the true solution.

11 indicates that the SpecKF would be more robust than the EnKF for a wider range of measurement errors.

Finally, we investigate the effect of the number of measurements for the SpecKF and the EnKF. Figure 14 shows the relative error for the original CO_2 problem as well as two more cases with 60 and 120 measurements, respectively. The number of ensemble members is kept constant at 58. Consistent with the literature, as the number of measurements increases beyond the number of the ensemble members, the estimation problem becomes ill-defined and the error increases. This illustrates that the number of ensemble members in the EnKF method should increase as the number of measurements increases. Therefore, the computational cost of both methods increase, as the number of measurements increases. However, it is not obvious for the EnKF how to choose the number of ensembles as the number of measurements increases. This affects the reliability and cost of applying EnKF. In contrast, for the SpecKF the increase in computational cost is exactly proportional to the number of measurements, and the increase in measurements

Figure 13: Comparison of true saturation with the SpecKF and the EnKF estimations for the case with small measurement error. First row) the SpecKF estimation for saturation and second row) the EnKF estimation for saturation with the same computational cost.

Figure 14: Comparison of the relative error of EnKF and SpecKF saturation estimation with respect to the true solution for three different number of measurements.

does not affect the reliability of the estimation. While we recognize that the above performance comparison may be specific to the application at hand,

non-linearities, parameterization of the filters etc, our results shows the reliability and robustness of the SpecKF, specifically in comparison to the EnKF, as the number of measurements increases.

4. Conclusions

This paper presented the Spectral Kalman Filter, a new Kalman Filter variant that can be effectively used for data assimilation in non-linear dynamical systems with a large number of unknowns. The computational cost of the SpecKF is dramatically reduced compared to that of the original Extended Kalman Filter, as it scales linearly (as opposed to quadratically) with the number of unknowns and makes the estimation of states in excess of thousands of unknowns feasible.

The SpecKF algorithm was validated for a linear diffusion problem, and it was shown that for a small time step between data assimilations the SpecKF gives results nearly as accurate as the full Kalman Filter. Our error analysis indicated that the SpecKF also provides accurate estimates of the uncertainty at the measurement points. Also, we demonstrated that the uncertainty can be obtained accurately at a specified location just by adding a control point to the interested location. Alternatively, a low-rank approximation of the full covariance matrix can be obtained. Since, this approximation is a post processing step, it does not affect the estimation with SpecKF.

Comparison of the SpecKF to the EnKF for both the linear and the non-linear benchmark problems showed that the SpecKF provides similar accuracy for the same or lower computational cost than the EnKF. The SpecKF is advantageous for non-linear estimation since its accuracy depends on con-

553 trollable, physically meaningful parameters, as opposed to the ensemble size
 554 of the EnKF, which is typically determined in an ad-hoc manner. In essence,
 555 the SpecKF does not approximate the covariance P , leading to consistent
 556 accuracy in estimation for non-linear problems. Also, for a large number of
 557 measurements, the EnKF requires an increase (also determined heuristically)
 558 in the number of ensemble members, while the SpecKF does not require any
 559 further adjustment. Similarly, when the noise in the measurements is small
 560 compared to the noise in the forward model, the accuracy of EnKF estimation
 561 will decrease, while the SpecKF has a consistent accuracy.

562 The SpecKF is a promising method for large-scale data assimilation, that
 563 does not suffer from the drawbacks of existing fast EKF variants. Future
 564 work will evaluate the SpecKF on field-scale problems and investigate the
 565 effects of non-linearity on the performance of the filter.

566 5. Acknowledgments

567 This material is based upon work supported by "US Department of
 568 Energy, National Energy Technology Laborator" (DOE, NETL) under the
 569 award DE-FE0009260: "An Advanced Joint Inversion System for CO_2 Stor-
 570 age Modeling with Large Data Sets for Characterization and Real-Time Mon-
 571 itoring", as well as by the NSF award DMS-1228275 "Large-scale stochastic
 572 inverse and data assimilation problems with applications to subsurface imag-
 573 ing".

6. Appendix

A. SpecKF algorithm

The SpecKF achieves computational performance by storing and updating small cross covariance matrices instead of the large covariance matrix. To do that, first the forecast and analysis P matrix (Eqns. (4b) and (4e) respectively) are multiplied with H^T .

$$P_{k+1}^f H^T = F_{k+1} P_k^a F_{k+1}^T H^T + Q_{k+1} H^T \quad (\text{A.1})$$

$$P_{k+1}^a H^T = P_{k+1}^f H^T - K_{k+1} H_{k+1} P_{k+1}^f H^T \quad (\text{A.2})$$

Next, the matrices T and P^H are defined as $T_{k+1} = P_{k+1}^f H^T$ and $P_{k+1}^H = P_{k+1}^a H^T$. Writing the recurrence in terms of T and P^H gives:

$$T_{k+1} = F_{k+1} P_k^a F_{k+1}^T H^T + Q_{k+1} H^T \quad (\text{A.3})$$

$$P_{k+1}^H = T_{k+1} - K_{k+1} H_{k+1} T_{k+1} \quad (\text{A.4})$$

Eq. (A.3) involves the term $P_k^a F_{k+1}^T H^T$, which is an expensive dense matrix-matrix multiplication. To avoid performing this multiplication, the recurrence is modified by multiplying Eq. (4e) by $F_{k+1}^T H^T$:

$$\begin{aligned} P_{k+1}^a F^T H^T &= (I - K_{k+1} H) P_{k+1}^f F^T H^T \\ &= (I - K_{k+1} H) (F P_k^a F^T F^T H^T + Q F^T H^T) \end{aligned} \quad (\text{A.5})$$

The next objective is to break the expensive term $P F^T F^T H^T$ of Eq. (A.5) into smaller terms that can be obtained by a recurrence, as opposed to an expensive matrix-matrix multiplication. Denote now:

$$\Delta_k = F_k - I$$

We assume that Δ_k is small, for example of order $\mathcal{O}(dt)$. This estimate holds by doing a Taylor expansion assuming that F_k is differentiable with respect to time. Rewriting $PF^T F^T H^T$ using this discretization, we obtain:

$$\begin{aligned}
P_k F^T F^T H^T &= P_k (I + \Delta_k)^T (I + \Delta_k)^T H^T \\
&= P_k (I + 2\Delta_k^T + (\Delta_k^2)^T) H^T \\
&= P_k (2F^T - I + (\Delta_k^2)^T) H^T \\
&= 2P_k F^T H^T - P_k H^T + P_k (\Delta_k^2)^T H^T \\
&= 2P_k^F - P_k^H + \mathcal{O}(dt^2)
\end{aligned} \tag{A.6}$$

Substituting Eq. (A.6) in Eq. (A.5) results in:

$$P_{k+1}^F = (I - K_{k+1}H)(2F P_k^F - F P_k^H + Q F^T H^T) + \mathcal{O}(dt^2) \tag{A.7}$$

Following this process, we have obtained a recurrence for all cross covariance matrices, at the expense of adding a second order error in time.

$$s_{k+1}^f = F_{k+1} s_k^a \tag{A.8a}$$

$$T_{k+1} = F_{k+1} P_k^F + Q_{k+1} H^T \tag{A.8b}$$

$$K_{k+1} = T_{k+1} (H_{k+1} T_{k+1} + R_{k+1})^{-1} \tag{A.8c}$$

$$s_{k+1}^a = s_{k+1}^f + K_{k+1} (y_{k+1} - H_{k+1} s_{k+1}^f) \tag{A.8d}$$

$$P_{k+1}^F = (I - K_{k+1}H)(2F P_k^F - F P_k^H + Q F^T H^T) + \mathcal{O}(dt^2) \tag{A.8e}$$

$$P_{k+1}^H = T_{k+1} - K_{k+1} H_{k+1} T_{k+1} \tag{A.8f}$$

However, in this process, there is no low-rank approximation of the covariance matrix P . As a result, the SpecKF does not suffer from rank deficiency

problems, a possible problem for low-rank Kalman Filters. The only error that is added in the SpecKF is a second order error in time that can be controlled by decreasing the time step dt between assimilations steps.

If further reduction in the error is desired, the SpecKF algorithm can be modified to obtain higher orders of accuracy by writing appropriate recurrences for PH^T , $PF_t^T H^T$, $P(F_t^T)^2 H^T$, $\dots$, $P(F_t^T)^s H^T$, which are $\mathcal{O}(dt^{s+1})$ accurate. In order to get $((dt)^{s+1})$ order of accuracy, the last step of the original KF algorithm should be multiplied with H^T , $F_t^T H^T$, $(F_t^T)^2 H^T$, $\dots$, $(F_t^T)^s H^T$, to give a recurrence equation for all PH^T , $PF_t^T H^T$, $P(F_t^T)^2 H^T$, $\dots$, $P(F_t^T)^s H^T$ terms. The only term that is problematic is $P(F_t^T)^{s+1} H^T$, which always appears in the last equation of the recurrence. Modifications as described above can be used to approximate this term as a sum of terms that are already available from the previous time steps of the Kalman Filter. Rewriting F as $\Delta + I$, we have:

$$P(F^T)^{s+1} H^T = P((\Delta + I)^{s+1})^T H^T \quad (\text{A.9})$$

$$\begin{aligned} &= PH^T + \alpha_1 PF^T H^T + \dots \\ &+ \alpha_s P(F^s)^T H^T \\ &+ \alpha_{s+1} P(\Delta^{s+1})^T H^T \end{aligned} \quad (\text{A.10})$$

$$\begin{aligned} &= PH^T + \alpha_1 PF^T H^T + \dots \\ &+ \alpha_s P(F^s)^T H^T \\ &+ \mathcal{O}(dt^{s+1}) \end{aligned} \quad (\text{A.11})$$

where α_k is a constant; therefore we have all the first $s+1$ terms, and we can substitute $PH^T + \alpha_1 PF^T H^T + \dots + \alpha_s P(F^s)^T H^T$ instead of $P(F_t^T)^{s+1} H^T$ term, with an error $\mathcal{O}((dt)^{s+1})$.

Following the above procedure, the general SpecKF algorithm with $(dt)^{s+1}$ order of accuracy can be obtained:

$$\begin{aligned}
s_{k+1}^f &= F_{k+1} s_k^a \\
T_{k+1} &= F_{k+1} P_k^F + Q_{k+1} H^T \\
K_{k+1} &= T_{k+1} (H_{k+1} T_{k+1} + R_{k+1})^{-1} \\
s_{k+1}^a &= s_{k+1}^f + K_{k+1} (y_{k+1} - H_{k+1} s_{k+1}^f) \\
P_{k+1}^H &= T_{k+1} - K_{k+1} H_{k+1} T_{k+1} \\
P_{k+1} F^T H^T &= (I - K_{k+1} H) (F P_k (F^T)^2 H^T + Q F^T H^T) \\
&\vdots \\
P_{k+1} (F^T)^s H^T &= (I - K_{k+1} H) (F P_k H^T + \alpha_1 F P_k F^T H^T + \dots + \alpha_s F P_k (F^s)^T H^T + \\
&Q (F^T)^s H^T) + \mathcal{O}(dt^{s+1})
\end{aligned}$$

B. EnKF algorithm

The ensemble Kalman filter algorithm used in this paper follows [14] and is briefly shown here:

$$\begin{aligned}
s_{k+1,j}^f &= f_{k+1}(s_{k,j}^a) + w_{k+1,j} \quad j = 1, \dots, N \\
D_{k+1} &= [s_{k+1,1}^f - \bar{s}_{k+1}^f, s_{k+1,2}^f - \bar{s}_{k+1}^f, \dots, s_{k+1,N}^f - \bar{s}_{k+1}^f] \\
P_{k+1}^f &= \frac{1}{N-1} D_{k+1} D_{k+1}^T \\
K_{k+1} &= P_{k+1}^f H^T (H P_{k+1}^f H^T + R) \\
s_{k+1,j}^a &= s_{k+1,j}^f + K_{k+1} (y_{k+1} - H s_{k+1,j}^f + v_{k+1,j}) \\
s_{k+1}^a &= \frac{1}{N} \sum_{j=1}^N s_{k+1,j}^a
\end{aligned}$$

In this algorithm, we assume that w_{k+1} and v_{k+1} are zero mean normal random variables, with covariance Q_{k+1} and R_{k+1} .

1
2
3
4
5
6
7
8
9
620 *C. Estimation of the diagonal entries of the covariance matrix*

11 The SpecKF algorithm gives the posterior covariance only at measure-
12 ment points. However, we may need to know the uncertainty of the esti-
13 mates at locations that are not known a priori. We can obtain an estimate of
14 the full matrix P by assuming that the posterior covariance has a low-rank
15 representation:
16
17
18
19

$$P \approx USU^T$$

21
22
23 621 U is a known, preselected $m \times n$ matrix, i.e., we assume that P can be
24 approximated by a low-rank matrix with rank equal to the number of mea-
25 surements.
26
27 623

28 Then, the cross-covariance HPH^T , which is available in the SpecKF, can
29 be written as:
30
31

$$\begin{aligned} HPH^T &\approx HUSU^TH^T = (HU)S(HU)^T \\ &= (HUS^{1/2})(HUS^{1/2})^T \end{aligned}$$

32 Then, we perform a Cholesky decomposition of the known cross-covariance
33 $HPH^T = VV^T$. This gives:
34
35
36

$$V \approx HUS^{1/2}$$

37 Since V , H and U is known we can obtain an expression for $S^{1/2}$ and then
38 get the desired approximation of P .
39
40
41

$$S^{1/2} = (HU)^{-1}V$$

$$P \approx USU^T$$

51
52
53
54 624 This step assumes that HU is non-singular. H is a fat matrix with n rows,
55 while U is a thin matrix with n columns. For HU to be non-singular, we need
56
57
58
59
60
61
62
63
64
65

to assume that $\text{span}(U) \cap \text{null}(H) = \emptyset$. Geometrically, this means that none of the “important” components of P are in the null space of the observation matrix H , that is, if there are components of P ($= \text{span}(U)$) that are not observed, they cannot be reconstructed by this method.

7. References

- [1] Kalman, R. A., 1960. New approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82, 35-45.
<http://dx.doi.org/10.1115/1.3662552>
- [2] Lefebvre, T., and Bruyninckx, H., and De Schutter, J., 2004. Kalman Filters for nonlinear systems: a comparison of performance. *International Journal of Control*, 77, 639-653.
<http://dx.doi.org/10.1080/00207170410001704998>
- [3] Kushner, H. J., 1967. Approximation to optimal nonlinear filters. *IEEE Transactions on Automatic Control*, 12, 546-556.
<http://dx.doi.org/10.1109/TAC.1967.1098671>
- [4] Sorenson, H. W., and Stubberud, A. R., 1968. Non-linear filtering by approximation of the a posteriori density. *International Journal of Control*, 8, 33-51. <http://dx.doi.org/10.1080/00207176808905650>
- [5] Anderson, B. D. O. and Moore, J.B., 1979. Optimal filtering. *Application in nonlinear filtering*, 8, 193-221.
<http://dx.doi.org/10.1109/TSMC.1982.4308806>

- 647 [6] Nenna, V., Pidlisecky, A., and Knight, R., 2011. Application of an ex-
648 tended Kalman filter approach to inversion of time-lapse electrical resis-
649 tivity imaging data for monitoring recharge. Water Resources Research,
650 47, W10525, <http://dx.doi.org/10.1029/2010WR010120>
- 651 [7] Ghil, M., Cohn, S., Tavantzis, J., Bube, K., and Isaacson, E., 1981. Ap-
652 plication of estimation theory to numerical weather prediction. Dynamic
653 Meteorology: Data Assimilation Methods. Springer New York, 139-224.
654 http://dx.doi.org/10.1007/978-1-4612-5970-1_5
- 655 [8] Katul, G. and Wendroth, O., Parlange, M., and Puente, C., Folegatti,
656 M., and Nielsen, D., 1993. Estimation of in situ hydraulic conductivity
657 function from nonlinear filtering theory. Water Resources Research, 29,
658 1063-1070. <http://dx.doi.org/10.1029/92WR02593>
- 659 [9] Eigbe, U., Beck, M. B., Wheeler, H. S., and Hirano, F.,
660 1998. Kalman filtering in groundwater flow modelling: problems
661 and prospects. Stochastic Hydrology and Hydraulics, 12, 15-32.
662 <http://dx.doi.org/10.1007/s004770050007>
- 663 [10] Yeh, H. D., and Huang, Y. C., 2005. Parameter estimation for leaky
664 aquifers using the extended Kalman filter, and considering model and
665 data measurement uncertainties. Journal of Hydrology, 302, 28-45.
666 <http://dx.doi.org/10.1016/j.jhydrol.2004.06.035>
- 667 [11] Evensen, G., 1992. Using extended Kalman Filter with a multilayer
668 quasi-geostrophic model. Journal of Geophysical Research, 97, 17905-
669 17924. <http://dx.doi.org/10.1029/92JC01972>

- [12] Evensen, G., 1993. Open boundary condition for the extended Kalman Filter with a quasi-geostrophic model. *Journal of Geophysical Research*, 98, 16529-16546. <http://dx.doi.org/10.1029/93JC01365>
- [13] Evensen, G., 1994. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research*, 99, 10143-10162. <http://dx.doi.org/10.1029/94JC00572>
- [14] Evensen, G., 2003. The ensemble Kalman filter: theoretical formulation and practical implementation. *Ocean Dynamics*, 53, 343-367. <http://dx.doi.org/10.1007/s10236-003-0036-9>
- [15] Burgers, G., van Leeuwen, P. J., and Evensen, G., 1998. Analysis scheme in the ensemble Kalman filter. *Monthly Weather Review*, 126, 1719-1724. [http://dx.doi.org/10.1175/1520-0493\(1998\)126%3C1719:ASITEK](http://dx.doi.org/10.1175/1520-0493(1998)126%3C1719:ASITEK)
- [16] Anderson, J. L., 2001. An ensemble adjustment Kalman filter for data assimilation. *Monthly Weather Review*, 129, 2884-2903. [http://dx.doi.org/10.1175/1520-0493\(2001\)129%3C2884:AEAKFF%3E2.0.CO;2](http://dx.doi.org/10.1175/1520-0493(2001)129%3C2884:AEAKFF%3E2.0.CO;2)
- [17] Houtekamer, P.L., and Mitchell, H.L., 1998. Data assimilation using an ensemble Kalman filter technique. *Monthly Weather Review*, 126, 796-811. [http://dx.doi.org/10.1175/1520-0493\(1998\)126<0796:DAUAEK>2.0.CO;2](http://dx.doi.org/10.1175/1520-0493(1998)126<0796:DAUAEK>2.0.CO;2)
- [18] Liu, N., and D. S. Oliver, 2005. Ensemble Kalman filter for automatic

- history matching of geologic facies. *Journal of Petroleum Science and Engineering*, 47, 147-161. <http://dx.doi.org/10.1016/j.petrol.2005.03.006>
- [19] Evensen, G., 2007. Data Assimilation: The Ensemble Kalman Filter. *Ensemble methods*, 9, 119-137.
- [20] Kepert, J. D., 2004. On ensemble representation of the observation-error covariance in the ensemble Kalman Filter. *Ocean Dynamics*, 54, 561-9. <http://dx.doi.org/10.1007/s10236-004-0104-9>
- [21] Aanonsen, S. I., Naevdal, G., Oliver, D. S., Reynolds, A. C., and Valles, B., 2008. The ensemble Kalman filter in reservoir engineering—a Review. *SPE Journal*, 14, 393-412. <http://dx.doi.org/10.2118/117274-PA>
- [22] Furrer, R., and Bengtsson, T., 2007. Estimation of high-dimensional prior and posterior covariance matrices in Kalman filter variants. *Journal of Multivariate Anal*, 98, 227-255. <http://dx.doi.org/10.1016/j.jmva.2006.08.003>
- [23] Houtekamer, P. L., and Mitchell, H. L., 2001. A sequential ensemble Kalman filter for atmospheric data assimilation. *Monthly Weather Review*, 129, 123-137. [http://dx.doi.org/10.1175/1520-0493\(2001\)129%3C0123:ASEKFF%3E2.0.CO;2](http://dx.doi.org/10.1175/1520-0493(2001)129%3C0123:ASEKFF%3E2.0.CO;2)
- [24] Gaspari, G., and Cohn, S. E., 1999. Construction of correlation functions in two and three dimensions. *Journal of the Royal Meteorological Society*, 125, 723-757. <http://dx.doi.org/10.1002/qj.49712555417>
- [25] Kitanidis, P. K., 2015. Compressed State Kalman Filter for large systems. *Advance in Water Resources*, 76, 120-126.

- [26] Li, J. Y., Ambikasaran, S., Darve, E. F., and Kitanidis, P. K., 2014. A Kalman filter powered by H^2 matrices for quasi continuous data assimilation problems. Water Resources Research, under submission. <http://dx.doi.org/10.1002/2013WR014607>
- [27] Li, J. Y., Darve, E. F., and Kitanidis, P. K., 2014. Compressed state Kalman filter for CO_2 reservoir monitoring. Water Resources Research, under submission.
- [28] Pham, D. T., Verron, J., and Roubaud, C. M., 1998. A singular evolutive extended Kalman filter for data assimilation in oceanography. Journal of Marine Systems, 16, 323-340. [http://dx.doi.org/10.1016/S0924-7963\(97\)00109-7](http://dx.doi.org/10.1016/S0924-7963(97)00109-7)
- [29] Brasseur, P., Ballabrera, J., and Verron, J., 1999. Assimilation of altimetric data in the mid-latitude oceans using the Singular Evolutive Extended Kalman filter with an eddy-resolving, primitive equation model. Journal of Marine Systems, 22, 269-294. [http://dx.doi.org/10.1016/S0924-7963\(99\)00044-5](http://dx.doi.org/10.1016/S0924-7963(99)00044-5)
- [30] Nowak, W., Tenkleve, S., and Cirpka, O.A., 2003. Efficient computation of linearized cross-covariance and auto-covariance matrices of interdependent quantities. Mathematical Geology, 35, 1, 53-66. <http://dx.doi.org/10.1023/A%3A1022365112368>
- [31] Fong, W. and Darve, E., 2009. The black-box fast multipole method. Journal of Computational Physics, 228, 8712-8725. <http://dx.doi.org/10.1016/j.jcp.2009.08.031>

- [32] Saibaba, A. K., and Kitanidis, P. K., 2012. Efficient methods for large-scale linear inversion using a geostatistical approach. *Water Resources Research*, 48, 5, W05522. <http://dx.doi.org/10.1029/2011WR011778>
- [33] Saibaba, A.K., Ambikasaran, S., Li, J.Y., Kitanidis, P.K., and Darve, E., 2012. Application of hierarchical matrices to linear inverse problems in geostatistics. *OGST Revue d'IFP Energies Nouvelles*, 67, 5, 857-875. <http://dx.doi.org/10.2516/ogst/2012064>
- [34] Ambikasaran, S. and Saibaba, A. K. and Darve, E.F. and Kitanidis, P. K., 2013. Fast algorithms for Bayesian Inversion, *The IMA Volumes in Mathematics and its Applications*, vol, 156. http://dx.doi.org/10.1007/978-1-4614-7434-0_5
- [35] Kitanidis, P. K., and Lee, J. Y., 2013. Principal component geostatistical approach for large dimensional inverse problems. *Water Resources Research*, 50, 5428-5443. <http://dx.doi.org/10.1002/2013WR014630>
- [36] Pruess, K., 1991. TOUGH2: A general-purpose numerical simulator for multiphase fluid and heat flow. NASA STI/Recon Technical Report N, 92, 14316. <http://dx.doi.org/10.2172/5212064>
- [37] Pruess, K., Oldenburg, C.M., and Moridis, G., 1999. TOUGH2 User's Guide, Version 2.0. Lawrence Berkeley National Laboratory, Report LBNL-43134, 197 pp.
- [38] Finsterle, S., Sonnenthal, E. L., and Spycher, N., 2014. Advances in subsurface modeling using the {TOUGH} suite of simulators. *Computers & Geosciences*, 65, 2-12. <http://dx.doi.org/10.1016/j.cageo.2013.06.009>

- [39] Pruess, K., and Spycher, N., 2007. ECO2N - A fluid property module for the TOUGH2 code for studies of CO₂ storage in saline aquifers. *Energy Conversion and Management*, 48, 1761-1767. <http://dx.doi.org/10.1016/j.enconman.2007.01.016>
- [40] Zhang, K., Moridis, G., and Pruess, K., 2011. TOUGH+CO₂: A multiphase fluid-flow simulator for CO₂ geologic sequestration in saline aquifers. *Computers & Geosciences*, 37, 714-723. <http://dx.doi.org/10.1016/j.cageo.2010.09.011>
- [41] Rutqvist, J., Birkholzer, J., Cappa, F., and Tsang, C. F., 2007. Estimating maximum sustainable injection pressure during geological sequestration of CO₂ using coupled fluid flow and geomechanical fault-slip analysis. *Energy Conversion and Management*, 48, 1798-1807. <http://dx.doi.org/10.1016/j.enconman.2007.01.021>
- [42] Tran Ngoc, T. D., Doughty, C., Lefebvre, R. and Malo, M. 2013. Injectivity of carbon dioxide in the St. Lawrence Platform, Quebec (Canada): A sensitivity study. *Greenhouse Gas Sci Technol*, 3: 516540. <http://dx.doi.org/10.1002/ghg.1387>
- [43] Pruess, K., and Nordbotten, J., 2011. Numerical simulation studies of the long term evolution of a CO₂ plume in a saline aquifer with a sloping caprock. *Transport in Porous Media*, 90, 135-151. <http://dx.doi.org/10.1007/s11242-011-9729-6>
- [44] Doughty, C., 2010. Investigation of CO₂ plume behavior for a large scale

- 783 pilot test of geologic carbon storage in a saline formation. Transport in
784 Porous Media, 82, 49-76. <http://dx.doi.org/10.1007/s11242-009-9396-z>
- 785 [45] Doughty, C., 2007. Modeling geologic storage of carbon diox-
786 ide: Comparison of non-hysteretic and hysteretic characteristic
787 curves. Energy Conversion and Management, 48, 6, 1768-1781.
788 <http://dx.doi.org/10.1016/j.enconman.2007.01.022>
- 789 [46] Giorgis, T., Carpita, M., and Battistelli, A., 2007. 2D modeling
790 of salt precipitation during the injection of dry CO_2 in a depleted
791 gas reservoir. Energy Conversion and Management, 48, 1816-1826.
792 <http://dx.doi.org/10.1016/j.enconman.2007.01.012>
- 793 [47] Oldenburg, C.M., and Doughty, C., 2011. Injection, flow, and mixing of
794 CO_2 in porous media with residual gas. Transport in Porous Media, 90,
795 201-218. <http://dx.doi.org/10.1007/s11242-010-9645-1>
- 796 [48] Doughty, C., and Freifeld, B. M., 2013. Modeling CO_2 in-
797 jection at Cranfield, Mississippi: Investigation of methane and
798 temperature effects. Greenhouse Gas Sci Technol, 3, 475490.
799 <http://dx.doi.org/10.1002/ghg.1363>
- 800 [49] Humez, P., Audigane, P., Lions, J., Chiaberge, C., and Bellenfant, G.,
801 2011. Modeling of CO_2 leakage up through an abandoned well from deep
802 saline aquifer to shallow fresh groundwaters. Transport in Porous Media,
803 90, 153-181. <http://dx.doi.org/10.1007/s11242-011-9801-2>
- 804 [50] Jung, Y., Zhou, Q., and Birkholzer, J.T., 2013. Early detec-
805 tion of brine and CO_2 leakage through abandoned wells us-

- ing pressure and surface-deformation monitoring data: Concept
and demonstration. *Advances in Water Resources*, 62, 555-569.
<http://dx.doi.org/10.1016/j.advwatres.2013.06.008>
- [51] Tong, F., Niemi, A., Yang, Z., Fagerlund, F., Licha, T. and
Sauter, M., 2013. A numerical model of tracer transport in a
non-isothermal two-phase flow system for CO₂ geological stor-
age characterization. *Transport in Porous Media*, 98, 173-192.
<http://dx.doi.org/10.1016/j.chemgeo.2007.06.009>
- [52] Yang, Z. and Tian, L. and Niemi, A. and Fagerlund, F., 2013. Upscaling
of the constitutive relationships for CO₂ migration in multimodal hetero-
geneous formations. *International Journal of Greenhouse Gas Control*,
19, 743-755. <http://dx.doi.org/10.1016/j.ijggc.2012.11.015>
- [53] Mualem, Y., 1976. A new model for predicting the hydraulic conductiv-
ity of unsaturated porous media. *Water Resources Research*, 12, 57-64.
<http://dx.doi.org/10.1029/WR012i003p00513>
- [54] van Genuchten, M. Th., 1980. A closed-form equation for predicting the
hydraulic conductivity of unsaturated soils. *Soil Sci. soc.*, 44, 892-898.
<http://dx.doi.org/10.2136/sssaj1980.03615995004400050002x>
- [55] Wen, X.-H., and Chen, W.H., 2006. Real-time reservoir model updating
using ensemble Kalman Filter with confirming option. *SPE Journal*, 11,
431-442. <http://dx.doi.org/10.2118/92991-PA>