L_P REGULARIZATION FOR ENSEMBLE KALMAN INVERSION*

YOONSANG LEE†

Abstract. Ensemble Kalman inversion (EKI) is a derivative-free optimization method that lies between the deterministic and the probabilistic approaches for inverse problems. EKI iterates the Kalman update of ensemble-based Kalman filters, whose ensemble converges to a minimizer of an objective function. EKI regularizes ill-posed problems by restricting the ensemble to the linear span of the initial ensemble, or by iterating regularization with early stopping. Another regularization approach for EKI, Tikhonov EKI, penalizes the objective function using the l_2 penalty term, preventing overfitting in the standard EKI. This paper proposes a strategy to implement $l_p, 0 < p \leq 1$, regularization for EKI to recover sparse structures in the solution. The strategy transforms a l_p problem into a l_2 problem, which is then solved by Tikhonov EKI. The transformation is explicit, and thus the proposed approach has a computational cost comparable to Tikhonov EKI. We validate the proposed approach's effectiveness and robustness through a suite of numerical experiments, including compressive sensing and subsurface flow inverse problems.

Key words. inverse problems, ensemble Kalman inversion, regularization, sparsity

AMS subject classifications. 65J20, 65C05, 35Q93, 49M41

1. Introduction.

A wide range of problems in science and engineering are formulated as inverse problems. Inverse problems aim to estimate a quantity of interest from noisy, imperfect observation or measurement data, such as state variables or a set of parameters that constitute a forward model. Examples include deblurring and denoising in image processing [15], recovery of permeability in subsurface flow using pressure fields [27], and training a neural network in machine learning [16, 23] to name a few. In this paper, we consider the inverse problem of finding $u \in \mathbb{R}^N$ from measurement data $y \in \mathbb{R}^m$ where u and y are related as follows

$$y = G(u) + \eta. \tag{1.1}$$

Here $G : \mathbb{R}^N \to \mathbb{R}^m$ is a forward model that can be nonlinear and computationally expensive to solve, for example, solving a PDE problem. The last term η is a measurement error. The measurement error is unknown in general, but we assume that it is drawn from a known probability distribution, a Gaussian distribution with mean zero and a known covariance Γ. By assuming that the forward model G and the observation covariance Γ are known, the unknown variable u is estimated by solving an optimization problem

$$\operatorname*{argmin}_{u \in \mathbb{R}^N} \frac{1}{2} \|y - G(u)\|_\Gamma^2, \tag{1.2}$$

where $\|\cdot\|_\Gamma$ is the norm induced from the inner product using the inverse of the covariance matrix Γ, that is $\|a\|_\Gamma^2 = \langle a, \Gamma^{-1} a \rangle$ for the standard inner product $\langle , \rangle$ in $\mathbb{R}^m$.

Ensemble Kalman inversion (EKI), pioneered in the oil industry [27] and mathematically formulated in an application-neutral setting in [20], is a derivative-free method that lies between the deterministic and the probabilistic approaches for inverse problems. EKI's key feature is an iterative application of the Kalman update

*Submitted to the editors DATE.

Funding: This work was funded by NSF DMS-1912999 and ONR MURI N00014-20-1-2595.

†Department of Mathematics, Dartmouth College, NH (yoonsang.lee@dartmouth.edu).

of the ensemble-based Kalman filters [1, 13]. Ensemble-based Kalman filters are well known for their success in numerical weather prediction, stringent inverse problems involving high-dimensional systems. EKI iterates the ensemble-based Kalman update in which the ensemble mean converges to the solution of the optimization problem (1.2). EKI can be thought of as a least-squares method in which the derivatives are approximated from an empirical correlation of an ensemble [6], not from a variational approach. Thus, EKI is highly parallelizable without calculating the derivatives related to the forward or the adjoint problem used in the gradient-based methods.

Inverse problems are often ill-posed, which suffer from non-uniqueness of the solution and lack stability. Also, in the context of regression, the solution can show overfitting. A common strategy to overcome ill-posed problems is regularizing the solution of the optimization problem [3]. That is, a special structure of the solution from prior information, such as sparsity, is imposed to address ill-posedness. The standard EKI [20] implements regularization by restricting the ensemble to the linear span of the initial ensemble reflecting prior information. The ensemble-based Kalman update is known for that the ensemble remains in the linear span of the initial ensemble [25, 20]. Thus, the EKI ensemble always stays in the linear span of the initial ensemble, which regularizes the solution. Although this approach shows robust results in certain applications, numerical evidence demonstrates that overfitting may still occur [20]. As an effort to address the overfitting of the standard EKI, an iterative regularization method has been proposed in [21], which approximates the regularizing Levenberg-Marquardt scheme [18]. As another regularization approach using a penalty term to the objective function, a recent work called Tikhonov EKI (TEKI) [9] implements the Tikhonov regularization (which imposes a l_2 penalty term to the objective function) using an augmented measurement model that adds artificial measurements to the original measurement. TEKI's implementation is a straightforward modification of the standard EKI method with a marginal increase in the computational cost.

The regularization methods for EKI mentioned above address several issues of ill-posed problems, including overfitting. However, it is still an open problem to implement other types of regularizers, such as l_1 or total variation (TV) regularization. This paper aims to implement $l_p, 0 < p \leq 1$, regularization to recover sparse structures in the solution of inverse problems. In other words, we propose a highly-parallelizable derivative-free method that solves the following l_p regularized optimization problem

$$\underset{u \in X}{\operatorname{argmin}} \frac{\lambda}{2}\|u\|_p^p + \frac{1}{2}\|y - G(u)\|_\Gamma^2, \tag{1.3}$$

where $\|u\|_p$ is the l_p norm of u, i.e., $\sum_i^N |u_i|^p$, and λ is a regularization coefficient.

The proposed method's key idea is a transformation of variables that converts the l_p regularization problem to the Tikhonov regularization problem. Therefore, a local minimizer of the original l_p problem can be found by a local minimizer of the l_2 problem that is solved using the idea of Tikhonov EKI. As this transformation is explicit and easy to calculate, the proposed method's overall computational complexity remains comparable to the complexity of Tikhonov EKI. In general, a transformed optimization problem can lead to additional difficulties, such as change of convexity, increased nonlinearity, additional/missing local minima of the original problem, etc. [14]. We show that the transformation does not add or remove local minimizers in the transformed formulation. A work imposing sparsity in EKI has been reported recently [31]. The idea of this work is to use thresholding and a l_1 constraint to impose sparsity in the inverse problem solution. The l_1 constraint is further relaxed

by splitting the solution into positive and negative parts. The split converts the l_1 problem to a quadratic problem, while it still has a non-negativity constraint. On the other hand, our method does not require additional constraints by reformulating the optimization problem and works as a solver for the l_p regularized optimization problem (1.3).

This paper is structured as follows. Section 2 reviews the standard EKI and Tikhonov EKI. In section 3, we describe a transformation that converts the l_p regularization problem (1.3), $0 < p \leq 1$, to the Tikhonov (that is, l_2) regularization problem, and provide the complete description of the l_p regularized EKI algorithm. We also discuss implementation and computation issues. Section 4 is devoted to the validation of the effectiveness and robustness of regularized EKI through a suite of numerical tests. The tests include a scalar toy problem with an analytic solution, a compressive sensing problem to benchmark with a convex l_1 minimization method, and a PDE-constrained nonlinear inverse problem from subsurface flow. We conclude this paper in section 5, discussing the proposed method's limitations and future work.

2. Ensemble Kalman inversion. The l_p regularized EKI uses a change of variables to transform a l_p problem into a l_2 problem, which is then solved by the standard EKI using an augmented measurement model. This section reviews the standard EKI and the application of the augmented measurement model in Tikhonov EKI to implement l_2 regularization. The review is intended to be concise, delivering the minimal ideas for the l_p regularized EKI. Detailed descriptions of the standard EKI and the Tikhonov EKI methods can be found in [20] and [9], respectively.

2.1. Standard ensemble Kalman inversion. EKI incorporates an artificial dynamics, which corresponds to the application of the forward model to each ensemble member. This application moves each ensemble member to the measurement space, which is then updated using the ensemble Kalman update formula. The ensemble updated by EKI stays in the linear span of the initial ensemble [20, 25]. Therefore, by choosing an initial ensemble appropriately for prior information, EKI is regularized as the ensemble is restricted to the linear span of the initial ensemble. Under a continuous-time limit, when the operator G is linear, it is proved in [30] that EKI estimate converges to the solution of the following optimization problem

$$\operatorname*{argmin}_{u\in\mathbb{R}^N} \frac{1}{2}\|y - G(u)\|_\Gamma^2. \tag{2.1}$$

In this paper, we consider the discrete-time EKI in [20], which is described below.

Algorithm: standard EKI

Assumption: an initial ensemble of size K, $\{u_0^{(k)}\}_{k=1}^K$ from prior information, is given.
For $n = 1, 2, ...,$

1. Prediction step using the artificial dynamics:
   (a) Apply the forward model G to each ensemble member

$$g_n^{(k)} := G(u_{n-1}^{(k)}) \tag{2.2}$$

   (b) From the set of the predictions $\{g_n^{(k)}\}_{k=1}^K$, calculate the mean and covariances

$$\overline{g}_n = \frac{1}{K}\sum_{k=1}^K g_n^{(k)}, \tag{2.3}$$

$$C_n^{ug} = \frac{1}{K}\sum_{k=1}^{K}(u_n^{(k)} - \overline{u}_n) \otimes (g_n^{(k)} - \overline{g}_n),$$
$$C_n^{gg} = \frac{1}{K}\sum_{k=1}^{K}(g_n^{(k)} - \overline{g}_n) \otimes (g_n^{(k)} - \overline{g}_n), \tag{2.4}$$

where $\overline{u}_n$ is the mean of $\{u_n^{(k)}\}$, i.e., $\frac{1}{K}\sum_{k=1}^{K} u_n^{(k)}$.

2. Analysis step:
   (a) Update each ensemble member $u_n^{(k)}$ using the Kalman update

$$u_{n+1}^{(k)} = u_n^{(k)} + C_n^{ug}(C_n^{gg} + \Gamma)^{-1}(y_n^{(k)} - g_n^{(k)}), \tag{2.5}$$

   where $y_{n+1}^{(k)} = y + \zeta_{n+1}^{(k)}$ is a perturbed measurement using Gaussian noise $\zeta_{n+1}^{(k)}$ with mean zero and covariance Γ.
   (b) Compute the mean of the ensemble as an estimate for the solution

$$\overline{u}_{n+1} = \frac{1}{K}\sum_{k=1}^{K} u_n^{(k)} \tag{2.6}$$

Remark 2.1. The term $C_n^{ug}(C_n^{gg} + \Gamma)^{-1}$ in (2.5) is from the Kalman gain matrix. The standard EKI uses an extended space, $(u, G(u)) \in \mathbb{R}^{N+m}$, and then use the Kalman update for the extended space variable. However, as we need to update only u while $G(u)$ is subordinate to u, we have the update formula (2.5).

2.2. Tikhonov ensemble Kalman inversion. EKI is regularized through the initial ensemble reflecting prior information. However, there are several numerical evidence showing that EKI regularized only through an ensemble may have overfitting [20]. Among other approaches to regularize EKI, Tikhonov EKI [9] uses the idea of an augmented measurement to implement l_2 regularization, which is a simple modification of the standard EKI. For the original measurement y, the augmented measurement model extends y by adding the zero vector in $\mathbb{R}^N$, which yields an augmented measurement vector $z \in \mathbb{R}^{m+N}$

$$\text{augmented measurement vector: } z = (y, 0). \tag{2.7}$$

The forward model is also augmented to account for the augmented measurement vector, which adds the identity measurement

$$\text{augmented forward model: } F(u) = (G(u), u). \tag{2.8}$$

Using the augmented measurement vector and the model, Tikhonov EKI has the following inverse problem of estimating u from z

$$z = F(u) + \zeta. \tag{2.9}$$

Here ζ is a $m + N$-dimensional measurement error for the augmented measurement model, which is Gaussian with mean zero and covariance

$$\Sigma = \begin{pmatrix} \Gamma & 0 \\ 0 & \frac{1}{\lambda} I_N \end{pmatrix}, \tag{2.10}$$

for the $N \times N$ identity matrix I_N.

The mechanism enabling the l_2 regularization in Tikhonov EKI is the incorporation of the l_2 penalty term as a part of the augmented measurement model. From the orthogonality between different components in $\mathbb{R}^{m+N}$, we have

$$\begin{aligned} \frac{1}{2}\|z - F(u)\|_\Sigma^2 &= \frac{1}{2}\|y - G(u)\|_\Gamma^2 + \frac{1}{2}\|0 - u\|_{\frac{1}{\lambda}I_N}^2 \\ &= \frac{1}{2}\|y - G(u)\|_\Gamma^2 + \frac{\lambda}{2}\|u\|_2^2. \end{aligned} \tag{2.11}$$

Therefore, the standard EKI algorithm applied to the augmented measurement minimizes $\frac{1}{2}\|z - F(u)\|_\Sigma^2$, which equivalently minimizes the l_2 regularized problem.

3. l_p-regularization for EKI. This section describes a transformation that converts a $l_p, 0 < p \leq 1$, regularization problem to a l_2 regularization problem. l_p-regularized EKI (l_pEKI), which we completely describe in subsection 3.2, utilizes this transformation and solves the transformed l_2 regularization problem using the idea of Tikhonov EKI [9], the augmented measurement model.

3.1. Transformation of l_p regularization into l_2 regularization. For $0 < p \leq 1$, we define a function $\psi : \mathbb{R} \to \mathbb{R}$ given by

$$\psi(x) = \operatorname{sgn}(x)|x|^{\frac{p}{2}}, \quad x \in \mathbb{R}. \tag{3.1}$$

Here $\operatorname{sgn}(x)$ is the sign function of x, which has 1 for $x > 0$, 0 for $x = 0$, and -1 for $x < 0$. It is straightforward to check that ψ is bijective and has an inverse $\xi : \mathbb{R} \to \mathbb{R}$ defined as

$$\xi(x) = \operatorname{sgn}(x)|x|^{\frac{2}{p}}, \quad x \in \mathbb{R}. \tag{3.2}$$

For u in $\mathbb{R}^N$, we define a nonlinear map $\Psi : \mathbb{R}^N \to \mathbb{R}^N$, which applies ψ to each component of $u = (u_1, u_2, ..., u_N)$,

$$\Psi(u) = (\psi(u_1), \psi(u_2), ..., \psi(u_N)). \tag{3.3}$$

As ψ has an inverse, the map Ψ also has an inverse, say Ξ

$$\Xi(u) = \Psi^{-1}(u) = (\xi(u_1), \xi(u_2), ..., \xi(u_N)). \tag{3.4}$$

For $v = \Psi(u)$, it can be checked that for each $i = 1, 2, ..., N$,

$$|v_i|^2 = |\psi(u_i)|^2 = |u_i|^p,$$

and thus we have the following norm relation

$$\|v\|_2^2 = \|u\|_p^p. \tag{3.5}$$

This relation shows that the map $v = \Psi(u)$ converts the l_p-regularized optimization problem in u (1.3) to a l_2 regularized problem in v,

$$\operatorname*{argmin}_{v \in \mathbb{R}^N} \frac{\lambda}{2}\|v\|_2^2 + \frac{1}{2}\|y - \tilde{G}(v)\|_\Gamma^2, \tag{3.6}$$

where $\tilde{G}$ is the pullback of G by Ξ

$$\tilde{G} = G \circ \Xi. \tag{3.7}$$

A transformation between l_1 and l_2 regularization terms has already been used to solve an inverse problem in the Bayesian framework [32]. In the context of the randomize-then-optimize framework [2], the method in [32] draws a sample from a Gaussian distribution, which is then transformed to a Laplace distribution. As this method needs to match the corresponding densities of the variables (the original and the transformed variables) as random variables, the transformation involves calculations related to cumulative distribution functions. For the scalar case, $v \in \mathbb{R}$, the transformation from l_2 to l_1, denoted as gl, is given by

$$gl(v) = -\operatorname{sgn}(v) \log\left(1 - 2\left|\phi(v) - \frac{1}{2}\right|\right). \tag{3.8}$$

where $\phi(u)$ is the cumulative distribution function of the standard Gaussian distribution. Figure 1 shows the two transformations ξ (3.2) and gl (3.8); the former is based on the norm relation (3.5) and the latter is based on matching densities as random variables. We note that the transformation ξ has a region around 0 flatter than the

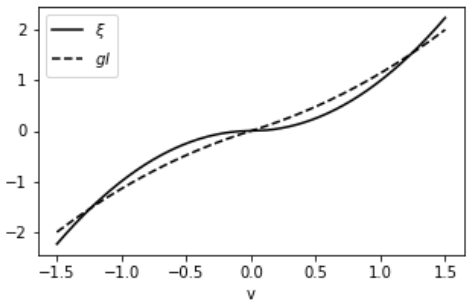


Fig. 1: ξ: transformation matching the norm relation (3.5), gl: transformation from Gaussian to Laplace distributions.

transformation gl, but ξ diverts quickly as v moves further away from 0. From this comparison, we expect that the flattened region of ξ plays another role in imposing sparsity by trapping the ensemble to the flattened area.

In general, a reformulation of an optimization problem using a transformation has the following potential issues [14]: i) the degree of nonlinearity may be significantly increased, ii) the desired minimum may be inadvertently excluded, or iii) an additional local minimum can be included. In [10], for a non-convex problem, it is shown that TEKI converges to an approximate local minimum if the gradient and Hessian of the objective function are bounded. It is straightforward to check that the transformed objective function has bounded gradient and Hessian if $0 < p \leq 1$ regardless of the convexity of the problem. Therefore, if we can show that the original and the transformed problems have the same number of local minima, then it is guaranteed to find a local minimum of the original problem by finding a local minimum of the transformed problem using TEKI. We want to note the importance of the sign function in defining ψ and ξ. The sign function is not necessary to satisfy the norm relation (3.5), but it is essential to make the transformation Ψ and its inverse Ξ bijective. Without being bijective, the transformed l_2 problem can have more or less local minima than the original problem.

The following theorem shows that the transformation does not add or remove local minima.

THEOREM 3.1. *For an objective function $J(u) : \mathbb{R}^N \to \mathbb{R}$, if u^* is a local minimizer of $J(u)$, $\Psi(u^*)$ is also a local minimizer of $\tilde{J}(v) = J \circ \Xi(v)$. Similarly, if v^* is a local minimizer of $\tilde{J}(v)$, then $\Xi(v^*)$ is also a local minimizer of $J(u) = \tilde{J} \circ \Psi(u)$.*

Proof. From the definition (3.3) and (3.4), Ψ and Ξ are continuous and bijective. Thus for $u \in \mathbb{R}^N$, both Ψ and Ξ map a neighborhood of $u \in \mathbb{R}^N$ to neighborhoods of $\Psi(u)$ and $\Xi(u)$, respectively. As u^* is a local minimizer, there exists a neighborhood $\mathcal{N}$ of u^* such that

$$J(u^*) \le J(w) \quad \text{for all } w \in \mathcal{N}. \tag{3.9}$$

Let $v = \Psi(u^*)$ and $\mathcal{M} := \Psi(\mathcal{N})$ that is a neighborhood of v. For any $w \in \mathcal{M}$, $\Xi(w) \in \mathcal{N}$ and thus we have

$$\tilde{J}(v) = J(\Xi(v)) = J(u) \le J(\Xi(w)) = \tilde{J}(w), \tag{3.10}$$

which shows that v is a local minimizer of $\tilde{J}$. The other direction is proved similarly by changing the roles of Ψ and Ξ and of J and $\tilde{J}$. ∎

We note that an insolated local minimizer can replace the local minimizer in the theorem. If there is a unique global minimizer of the l_p regularization problem (1.3), the theorem guarantees that we can find it by finding the global minimizer of the l_2 regularized problem (3.6).

COROLLARY 3.2. *For $0 < p \le 1$, if the l_p regularized optimization (1.3) has a unique global minimizer, say $u^\dagger$, the l_2 regularized optimization (3.6) also has a unique global minimizer. By finding the minimizer $u^\dagger$ of (3.6), say $v^\dagger$, $u^\dagger$ is given by*

$$u^\dagger = \Xi(v^\dagger). \tag{3.11}$$

3.2. Algorithm. l_p-regularized EKI (l_pEKI) solves the transformed l_2 regularization problem using the standard EKI with the augmented measurement model. For the current study's completeness to implement l_pEKI, this subsection describes the complete l_pEKI algorithm and discuss issues related to implementation. Note that the Tikhonov EKI (TEKI) part in l_pEKI is slightly modified to reflect the setting assumed in this paper. The general TEKI algorithm and its variants can be found in [9].

We assume that the forward model G and the measurement error covariance Γ are known, and measurement $y \in \mathbb{R}^m$ is given (and thus $z = (y, 0)$ is also given). We also fix the regularization coefficient λ and p. Under this assumption, l_pEKI uses the following iterative procedure to update the ensemble until the ensemble mean $\overline{v} = \frac{1}{K}\sum_{k=1}^{K} v^{(k)}$ converges.

Algorithm: l_p-regularized EKI

Assumption: an initial ensemble of size K, $\{v_0^{(k)}\}_{k=1}^K$, is given.
For $n = 1, 2, ...,$

1. Prediction step using the forward model:
   (a) Apply the augmented forward model F to each ensemble member

$$f_n^{(k)} := F(v_n^{(k)}) = (\tilde{G}(v_n^{(k)}), v_n^{(k)}) \tag{3.12}$$

(b) From the set of the predictions $\{f_n^{(k)}\}_{k=1}^K$, calculate the mean and covariances

$$\overline{f}_n = \frac{1}{K}\sum_{k=1}^{K} f_n^{(k)}, \tag{3.13}$$

$$\begin{aligned} C_n^{vf} &= \frac{1}{K}\sum_{k=1}^{K}(v_n^{(k)} - \overline{v}_n)\otimes(f_n^{(k)} - \overline{f}_n), \\ C_n^{ff} &= \frac{1}{K}\sum_{k=1}^{K}(f_n^{(k)} - \overline{f}_n)\otimes(f_n^{(k)} - \overline{f}_n) \end{aligned} \tag{3.14}$$

where $\overline{v}_n$ is the ensemble mean of $\{v_n^{(k)}\}$, i.e., $\dfrac{1}{K}\displaystyle\sum_{k=1}^{K} v_n^{(k)}$.

2. Analysis step:

(a) Update each ensemble member $v_n^{(k)}$ using the Kalman update

$$v_{n+1}^{(k)} = v_n^{(k)} + C_n^{vf}(C_n^{ff} + \Sigma)^{-1}(z_{n+1}^{(k)} - f_n^{(k)}), \tag{3.15}$$

where $z_{n+1}^{(k)} = z + \zeta_{n+1}^{(k)}$ is a perturbed measurement using Gaussian noise $\zeta_{n+1}^{(k)}$ with mean zero and covariance Σ.

(b) For the ensemble mean $\overline{v}_n$, the l_pEKI estimate, u_n, for the minimizer of the l_p regularization is given by

$$u = \Xi(\overline{v}_n). \tag{3.16}$$

Remark 3.3. In EKI and TEKI, the covariance of $\zeta_{n+1}^{(k)}$ can be set to zero so that all ensemble member uses the same measurement z without perturbations. In our study, we focus on the perturbed measurement using the covariance matrix Γ.

Remark 3.4. The above algorithm is equivalent to TEKI, except that the forward model G is replaced with the pullback of G by the transformation Ξ. In comparison with TEKI, the additional computational cost for l_pEKI is to calculate the Transformation $\Xi(v)$. In comparison with the standard EKI, the additional cost of l_pEKI, in addition to the cost related to the transformation, is the matrix inversion $(C_n^{gg}+\Sigma)^{-1}$ in the augmented measurement space $\mathbb{R}^{m+N}$ instead of a matrix inversion in the original measurement space $\mathbb{R}^m$. As the covariance matrices are symmetric positive definite, the matrix inversion can be done efficiently.

Remark 3.5. In l_pEKI, it is also possible to consider estimating u by transforming each ensemble member and take average of the transformed members, that is,

$$u = \frac{1}{K}\sum_{k=1}^{K}\Xi(v_n^{(k)}) \tag{3.17}$$

instead of (3.16). If the ensemble spread is large, these two approaches will make a difference. In our numerical tests in the next section, we do not incorporate covariance inflation. Thus the ensemble spread becomes relatively small when the estimate converges, and thus (3.16) and (3.17) are not significantly different. In this study, we use (3.16) to measure the performance of l_pEKI.

In recovering sparsity using the l_p penalty term, if the penalty term's convexity is not necessary, it is preferred to use a small $p < 1$ as a smaller p imposes stronger sparsity. The optimization problem (1.3) can be interpreted as a constrained optimization problem that minimizes the l_p term of u with a constraint related to the data. That is, the solution to the optimization problem is an intersection point of an l_p ball and an affine subspace [12]. For $p \leq 1$, the intersection point is expected to take place on the axes and thus lead to a sparse solution. In particular, it can be checked that a small $p < 1$ has a high chance to have the intersection point at the axes, which can impose stronger sparsity than a larger p. The transformation in l_pEKI works for any positive p, but the transformation can lead to an overflow for a small p; the function ξ depends on an exponent $\frac{2}{p}$ that becomes large for a small p. Therefore, there is a limit for the smallest p. In our numerical experiments in the next section, the smallest p is 0.7 in the compressive sensing test.

There is a variant of l_pEKI worth further consideration. In [30], a continuous-time limit of EKI has been proposed, which rescales $\Gamma \to h^{-1}\Gamma$ using $h > 0$ so that the matrix inversion $(C_n^{gg} + h^{-1}\Gamma)^{-1}$ is approximated by $h\Gamma^{-1}$ as a limit of $h \to 0$. In many applications, the measurement error covariance is assumed to be diagonal. That is, the measurement error corresponding to different components are uncorrelated. Thus the inversion Γ^{-1} becomes a cheap calculation in the continuous-time limit. The continuous-time limit is then discretized in time using an explicit time integration method with a finite time step. The latter is called 'learning rate' in the machine learning community, and it is known that an adaptive time-stepping to solve an optimization often shows improved results [11, 28]. The current study focuses on the discrete-time update described in (2.5) and we leave adaptive time-stepping for future work.

4. Numerical tests. We apply l_p-regularized EKI (l_pEKI) to a suite of inverse problems to check its performance in regularizing EKI and recovering sparse structures of solutions. The tests include: i) a scalar toy model where an analytic solution is available, ii) a compressive sensing problem to recover a sparse signal from random measurements of the signal, iii) an inverse problem in subsurface flow; estimation of permeability from measurements of hydraulic pressure field whose forward model is described by a 2D elliptic partial differential equation [8, 27]. In all tests, we run l_pEKI for various values of $p \leq 1$, and compare with the result of Tikhonov EKI. We analyze the results to check how effectively l_pEKI implements l_p regularization and recover sparse solutions. When available, we also compare l_pEKI with a l_1 convex minimization method. As quantitative measures for the estimation performance, we calculate the l_1 error of the l_pEKI estimates and the data misfit $\|y - G(u)\|_2$.

Several parameters are to be determined in l_pEKI to achieve robust estimation results, regularization coefficient λ, regularization power p, ensemble size, and its initialization. In this study, to focus on implementing l_p regularization for EKI without the effect of any particular strategy to choose the regularization coefficient, we find the coefficient by hand-tuning so that l_pEKI achieves the best result for a given p. In particular, we test λ that corresponds to $a \times 10^b$ where $a \in \{1, 2, ..., 9\}$ and $b \in \{-2, -1, ..., 3\}$ and select the result with the smallest l_1 error. We leave the l_pEKI performance investigation using other methods to choose λ, for example, cross-validation, as future work. In choosing the regularization power p, we also use a hand-tuning process. We gradually decrease p from 1 until l_pEKI diverges. Once we find the lower bound for p, we tune λ to obtain the best result for the lower bound p.

Ensemble initialization plays a role in regularizing EKI, restricting the estimate to the linear span of the initial ensemble. In our experiments, instead of tuning the initial ensemble for improved results, we initialize the ensemble using a Gaussian distribution with mean zero and a constant diagonal covariance matrix (the variance will be specified later for each test). As this initialization does not utilize any prior information, a sparse structure in the solution, we regularize the solution mainly through the l_p penalty term. For each test, we run 100 trials of l_pEKI through 100 realizations of the initial ensemble distribution and use the estimate averaged over the trials along with its standard deviation to measure the performance difference. We note that we tune λ for one trial and use the same λ for the other trials.

Regarding the ensemble size, for the scalar toy and the compressive sensing problems, we test ensemble sizes larger than the dimension of u, the unknown variable of interest. The purpose of a large ensemble size is to minimize the sampling error while we focus on the regularization effect of l_pEKI. To show the applicability of l_pEKI for high-dimensional problems, we also test a small ensemble size using the idea of multiple batches used in [29]. The multiple batch approach runs several batches where small magnitude components are removed after each batch. After removing small magnitude components from the previous batch, the ensemble is used for the next batch. The multiple batch approach enables a small ensemble size, 50 ensemble members, for the compressive sensing and the 2D elliptic inversion problems where the dimensions of u are 200 and 400, respectively.

In ensemble-based Kalman filters, covariance inflation is an essential tool to stabilize and improve the performance of the filters. In a connection with the inflation, an adaptive time-stepping has been investigated to improve the performance of EKI. Although the adaptive time-stepping can be incorporated in l_pEKI for performance improvements, we use the discrete version l_pEKI described in subsection 3.2 focusing on the effect of different types of regularization on inversion. We will report a thorough investigation along the line of adaptive time-stepping in another place.

4.1. A scalar toy problem. The first numerical test is a scalar problem for $u \in \mathbb{R}$ with an analytic solution. As this is a scalar problem, there is no effect of regularization from ensemble initialization, and we can see the effect from the l_p penalty term. The scalar optimization problem we consider here is the minimization of an objective function $J(u) = \frac{1}{4}|u|^p + \frac{1}{2}(1-u)^2$

$$\operatorname*{argmin}_{u\in\mathbb{R}} J(u) = \operatorname*{argmin}_{u\in\mathbb{R}} \frac{1}{4}|u|^p + \frac{1}{2}(1-u)^2. \tag{4.1}$$

This setup is equivalent to solving the optimization problem (1.3) using l_p regularization with $\lambda = 1/2$, where $y = 1$, $G(u) = u$, and η is Gaussian with mean zero and variance 1. Using the transformation $v = \Psi(u) = \psi(u) = \mathrm{sgn}(u)|u|^{\frac{p}{2}}$ defined in (3.1), l_pEKI minimizes a transformed objective function $\tilde{J}(v) = \frac{1}{4}|v|^2 + \frac{1}{2}(1-\mathrm{sgn}(v)|v|^{2/p})^2$

$$\operatorname*{argmin}_{v\in\mathbb{R}} \tilde{J}(v) = \operatorname*{argmin}_{v\in\mathbb{R}} \frac{1}{4}|v|^2 + \frac{1}{2}(1-\mathrm{sgn}(v)|v|^{2/p})^2, \tag{4.2}$$

which is an l_2 regularization of $\frac{1}{2}(1-\mathrm{sgn}(v)|v|^{\frac{2}{p}})^2$.

For $p = 1$, the first row of Figure 2 shows the objective functions of l_p (4.1) and the transformed l_2 (4.2) formulations. Each objective function has a unique global minimum without other local minima. The minimizers are $\frac{3}{4}$ and $\frac{\sqrt{3}}{2}$ for l_1 and l_2, respectively. We can check that the transformation does not add/remove local

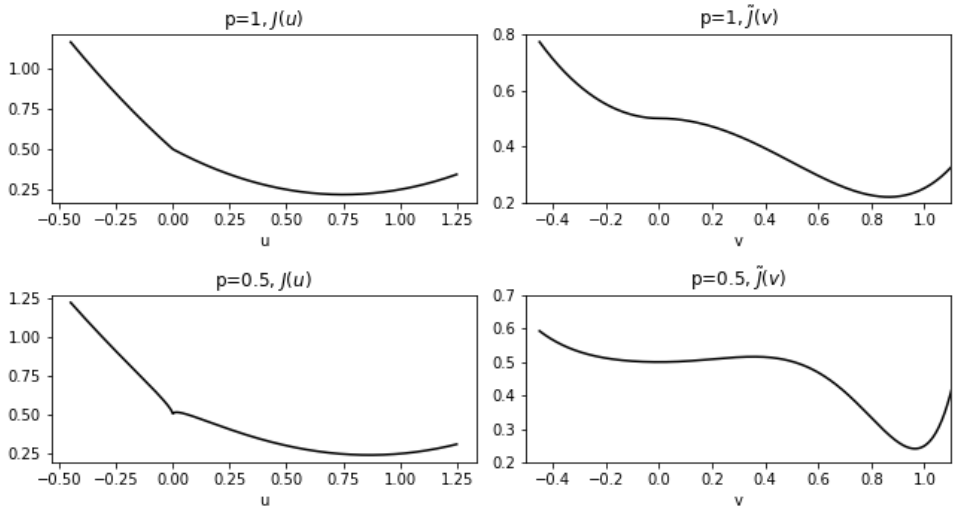


Fig. 2: Objective functions of (4.1) and (4.2) for $p = 1$ (first row) and $p = 0.5$ (second row).

minimizers, but the convexity of the objective function changes. The transformed objective function $\tilde{J}$ has an inflection point at $u = 0$, which is also a stationary point. Note that the original function has no other stationary points than the global minimizer.

When $p = 0.5$, a potential issue of the transformation can be seen explicitly. The original objective and the transformed objective functions are shown in the second row of Figure 2. Due to the regularization term with $p = 0.5$, the objective functions are non-convex and have a local minimizer at $u = v = 0$ in addition to the global minimizers. In the transformed formulation (bottom right of Figure 2), the objective function flattens around $v = 0$, which shows a potential issue of trapping ensemble members around $v = 0$. Numerical experiments show that if the ensemble is initialized with a small variance, the ensemble is trapped around $v = 0$. On the other hand, if the ensemble is initialized with a sufficiently large variance (so that some of the ensemble members are initialized out of the well around $v = 0$), l_pEKI shows convergence to the true minimizer, $v = 0.9304$ (or $u = 0.8656$) even when it is initialized around 0.

We use 100 different realizations for the ensemble initialization and each trial uses 50 ensemble members. The estimates at each iteration, which is averaged over different trials, are shown in Figure 3. For $p = 1$ (first row) and $p = 0.5$ (second row), the left and right columns show the results when the ensemble is initialized with mean 1 and 0, respectively. When $p = 1$ and initialized around 1, the ensemble estimate quickly converges to the true value 0.75 as the objective function is convex, and the initial guess is close to the true value. When $p = 0.5$, as the objective function is non-convex due to the regularization term, the convergence is slower than the $p = 1$ case. When the ensemble is initialized around 0 for $p = 0.5$, a local minimizer, the ensemble needs to be initialized with a large variance. Using variance 1, which is 10 times larger than 0.1, the variance for the ensemble initialization around 1, l_pEKI converges to the true value. The performance difference between different trials is

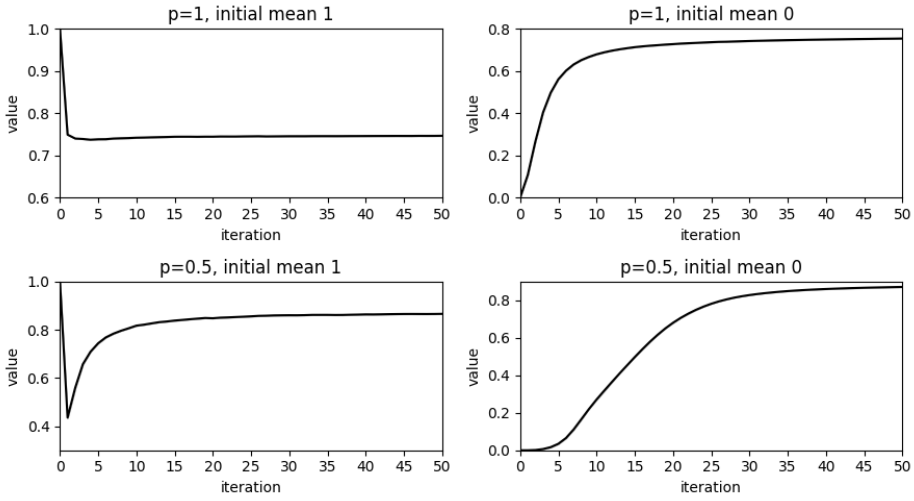


Fig. 3: Time series of l_pEKI estimate, $\xi(\overline{v}_n)$, which is averaged over 100 different trials.

marginal. The standard deviations of the estimate after 50 iterations are 6.62×10^{-3} ($p = 1$ initialized with 1), 7.95×10^{-3} ($p = 1$ initialized with 0), 8.79×10^{-3} ($p = 0.5$ initialized with 1), and 1.14×10^{-2} ($p = 0.5$ initialized with 0). As a reference, the estimate using the transformation (3.8) based on matching the densities of random variables converges to 0.71.

4.2. Compressive sensing. The second test is a compressive sensing problem. The true signal u is a vector in $\mathbb{R}^{200}$, which is sparse with only four randomly selected non-zero components (their magnitudes are also randomly chosen from the standard normal distribution.) The forward model $G : \mathbb{R}^{200} \to \mathbb{R}^{20}$ is a random Gaussian matrix of size 20×200, which yields a measurement vector in $\mathbb{R}^{20}$. The measurement y is obtained by applying the forward model to the true signal u polluted by Gaussian noise with mean zero and variance 0.01

$$y = Gu + \eta, \quad G \in \mathbb{R}^{20\times 200}, \eta \sim \mathcal{N}(0, 0.01). \tag{4.3}$$

As the forward model is linear, several robust methods can solve the sparse recovery problem, including the l_1 convex minimization method [4]. This test aims to compare the performance of l_pEKI for various p values, rather than to advocate the use of l_pEKI over other standard methods. As the forward model is linear and cheap to calculate, the standard methods are preferred over l_pEKI for this test.

We first use a large ensemble size, 2000 ensemble members, to run l_pEKI. The ensemble is initialized by drawing samples from a Gaussian distribution with mean zero and a diagonal covariance (which yields variance 0.1 for each component). For $p = 1$ and 0.7, the tuned regularization coefficients, λ, are 100 and 300. When $p = 2$, which corresponds to TEKI, the best result can be obtained using λ ranging from 10 to 200; we use the result of $\lambda = 50$ to compare with the other cases. For $p = 1$, we also compare the result of the convex l_1 minimization method using the interior point method using the Karush-Kuhn-Tucker condition [5] implemented in the Python library CVXOPT [26].

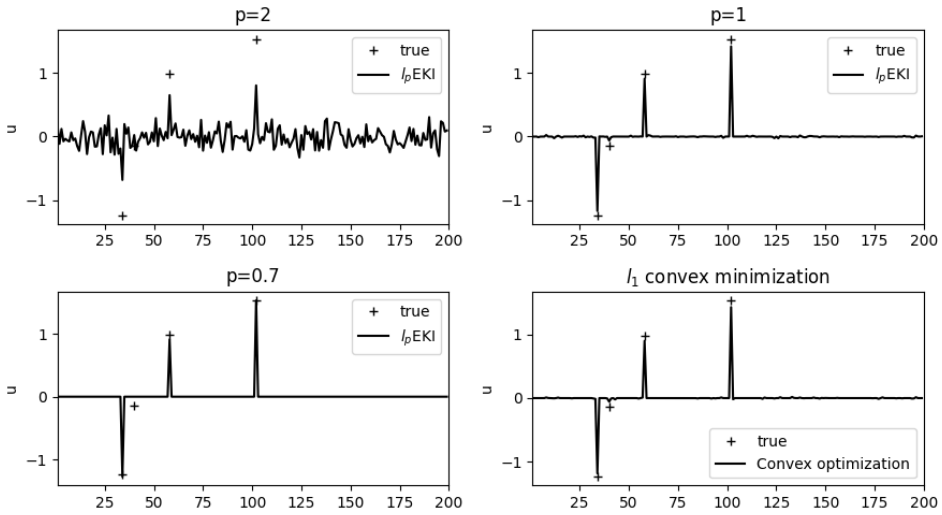


Fig. 4: Compressive sensing. Reconstruction of sparse signal using l_pEKI for p=2, 1, and 0.7. Ensemble size is 2000. The bottom right plot is the reconstruction using the convex l_1 minimization method. For the true signal, only the nonzero components are marked.

Figure 4 shows the l_pEKI estimates after 20 iterations averaged over 100 trials for $p = 2$ (top left), $p = 1$ (top right), and $p = 0.7$ (bottom left), along with the estimate by the convex optimization (bottom right). As it is well known in compressive sensing, l_2 regularization fails to capture the true signal's sparse structure. As p decreases to 1, l_pEKI develops sparsity in the estimate, comparable to the estimate of the convex l_1 minimization method. The slightly weak magnitudes of the three most significant components by l_pEKI improve as p decreases to 0.7. When $p = 0.7$, l_pEKI captures the correct magnitudes at the cost of losing the smallest magnitude component. The smallest magnitude component can be captured if the regularization coefficient λ decreases to 20 (see the left plot of Figure 5 for the l_pEKI estimate with $\lambda = 20$). However, this estimate also has several artificial non-zero components, which increases the l_1 error by about 15%. We note that the smallest magnitude component is challenging to capture; the magnitude is comparable to the measurement error $0.1 = \sqrt{0.01}$. When the measurement error variance decreases by a factor of 10, l_pEKI with $p = 0.7$ captures the smallest magnitude component with less significant artificial non-zero components (the right plot of Figure 5).

Another cost of using $p < 1$ to impose stronger sparsity than $p = 1$ is a slow convergence rate of l_pEKI. The time series of the l_1 estimation error and the data misfit of l_pEKI averaged over 100 trials are shown in Figure 6 alongside those of the convex optimization method. The results show that $p = 0.7$ converges slower than $p = 1$ (see Table 1 for the numerical values of the error and the misfit). Although there is a slowdown in convergence, it is worth noting that l_pEKI with $p = 0.7$ converges in a reasonably short time, 15 iterations, to achieve the best result. l_pEKI with $p = 2$ converges fast with the smallest data misfit. In this case, by combining many columns of G, l_pEKI makes a good approximation to the measurement error,

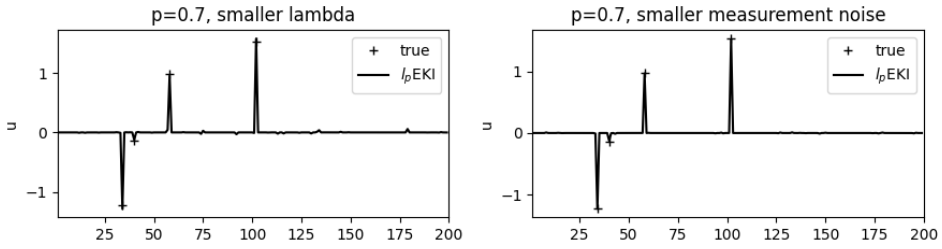


Fig. 5: l_pEKI estimates capturing the smallest magnitude component. Left: uses a smaller $\lambda = 20$. Right: uses a smaller measurement error variance 10^{-3}.

Method	l_1 error	data misfit
$p = 2$, ens size 2000	14.0802	0.0515
$p = 1$, ens size 2000	0.7848	0.8018
$p = 0.7$, ens size 2000	0.2773	1.2737
$p = 1$, ens size 50	1.6408	1.4095
$p = 0.7$, ens size 50	0.6027	1.8958
l_1 convex minimization	0.5623	0.9030

Table 1: Compressive sensing. l_pEKI estimate l_1 error and data misfit for $p = 2, 1$ and 0.7.

which yields a data misfit smaller than the actual norm of the measurement error 0.6014. In comparison, the other methods have misfits larger than the measurement norm. However, the l_2 regularization is not strong enough to impose sparsity in the estimate and yields the largest estimation error, which is 20 times larger than the case of $p = 1$. Note that the convex optimization method has the fastest convergence rate; it converges within three iterations and captures the four nonzero components with slightly smaller magnitudes than $p = 0.7$ for the three most significant components.

The ensemble size 2000 is larger than the dimension of the unknown vector u, 200. A large ensemble size can be impractical for a high-dimensional unknown vector. To see the applicability of l_pEKI using a small ensemble size, we use 50 ensemble members and two batches following the multiple batch approach [29]. The first batch runs 10 iterations, and all components whose magnitudes are less than 0.1 (the square root of the observation variance) are removed. The problem's size the second batch solves ranges from 30-45 (depending on a realization of the initial ensemble), which is then solved for another 10 iterations. The estimates using 50 ensemble members for $p = 1$ and $p = 0.7$ after two batch runs (i.e., 20 iterations) are shown in Figure 7. Compared with the large ensemble size case, the small ensemble size run also captures the most significant components at the cost of fluctuating components larger than the large ensemble size test. We note that the estimates are averaged over 100 trials, and thus there are components whose magnitudes are less than the threshold value 0.1 used in the multiple batch run.

As a measure to check the performance difference for different trials, Figure 8 shows the standard deviations of l_pEKI estimates for $p = 1$ and 0.7 after 20 iterations. The first row shows the results using 2000 ensemble members, while the second row

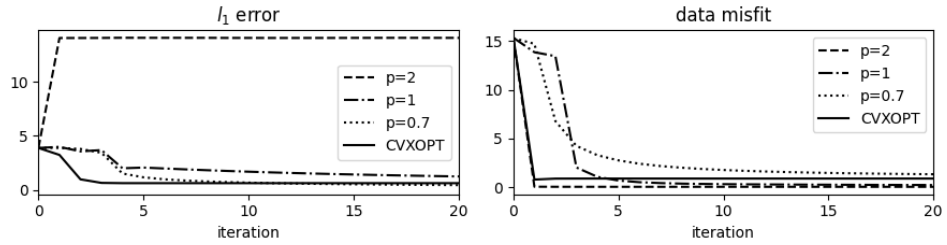


Fig. 6: Compressive sensing. l_1 error of the l_pEKI estimate and data misfit.

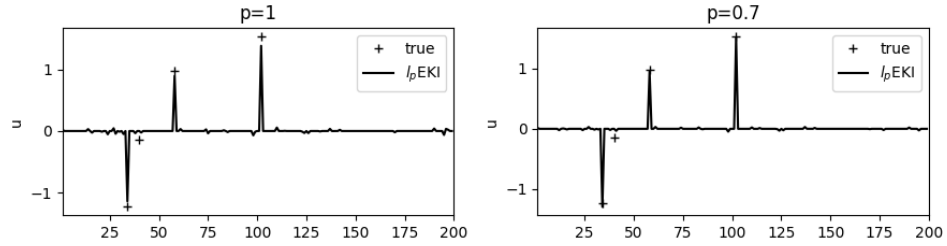


Fig. 7: Compressive sensing. Reconstruction of sparse signal using l_pEKI for p=1 and 0.7. Ensemble size is 50. For the true signal, only the nonzero components are marked.

shows the ones using 50 ensemble members. The standard deviations of the large ensemble size are smaller than those of the small ensemble size case as the large ensemble size has a smaller sampling error. In all cases, the standard deviations are smaller than 6% of the magnitude of the most significant components. In terms of p, the standard deviations of $p = 0.7$ are smaller than those of $p = 1$.

4.3. 2D elliptic problem. Next, we consider an inverse problem where the forward model is given by an elliptic partial differential equation. The model is related to subsurface flow described by Darcy flow in the two-dimensional unit square $(0, 1)^2 \subset \mathbb{R}^2$

$$-\nabla \cdot (k(x)\nabla p(x)) = f(x), \quad x = (x_1, x_2) \in (0, 1)^2. \tag{4.4}$$

The scalar field $k(x) > \alpha > 0$ is the permeability, and another field $p(x)$ is the piezometric head or the pressure field of the flow. For a known source term $f(x)$, the inverse problem estimates the permeability from measurements of the pressure field p. This model is a standard model for an inverse problem in oil reservoir simulations and has been actively used to measure EKI's performance and its variants, including TEKI [20, 9].

We follow the same setting used in TEKI [9] for the boundary conditions and the source term. The boundary conditions consist of Dirichlet and Neumann boundary conditions

$$p(x_1, 0) = 100, \frac{\partial p}{\partial x_1}(1, x_2) = 0, -k\frac{\partial p}{\partial x_1}(0, x_2) = 500, \frac{\partial p}{\partial x_2}(x_1, 1) = 0,$$

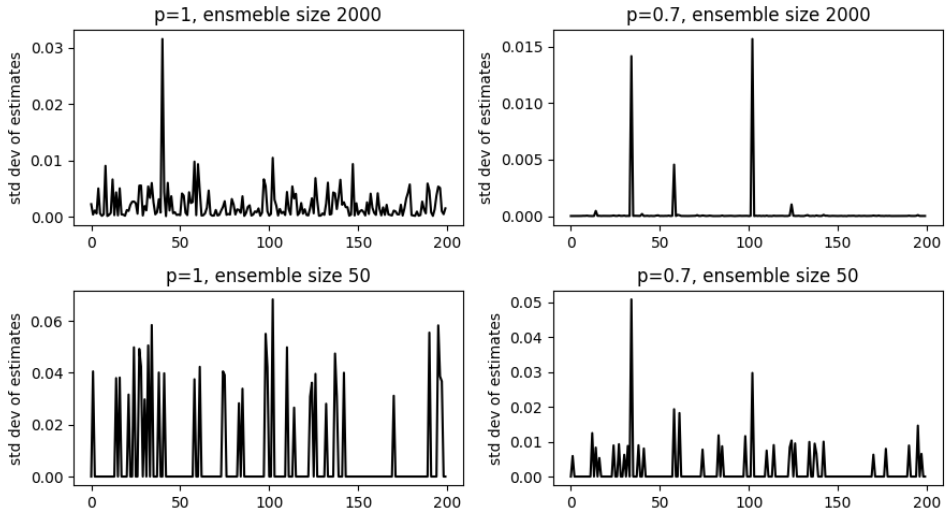


Fig. 8: Compressive sensing. Standard deviation of the estimates using 100 trials.

and the source term is piecewise constant

$$f(x_1, x_2) = \begin{cases} 0 & \text{if } 0 \le x_2 \le \frac{4}{6}, \\ 137 & \text{if } \frac{4}{6} < x_2 \le \frac{5}{6}, \\ 274 & \text{if } \frac{5}{6} < x_2 \le 1. \end{cases}$$

A physical motivation of the above configuration can be found in [8]. We use 15×15 regularly spaced points in $(0,1)^2$ to measure the pressure field with a small measurement error variance 10^{-6}. For a given k, the forward model is solved by a FEM method using the second-order polynomial basis on a 60×60 uniform mesh.

In addition to the standard setup, we impose a sparse structure in the permeability. We assume that the log permeability, $\log k$, can be represented by 400 components in the cosine basis $\phi_{ij} = \cos(i\pi x_1)\cos(j\pi x_2), i, j = 0, 1, ..., 19$,

$$\log k(x) = \sum_{i,j=0}^{19} u_{ij}\phi_{ij}(x), \tag{4.5}$$

where only six of $\{u_{ij}\}$ are nonzero. That is, we assume that the discrete cosine transform of $\log k$ is sparse with only 6 nonzero components out of 400 components. Thus, the problem we consider here can be formulated as an inverse problem to recover $u = \{u_{ij}\} \in \mathbb{R}^{400}$ (which has only six nonzero components) from a measurement $y \in \mathbb{R}^{225}$, the measurement of p at 15×15 regularly spaced points. In terms of sparsity reconstruction, the current setup is similar to the previous compressive sensing problem, but the main difference lies in the forward model. In this test, the forward model is nonlinear and computationally expensive to solve, where the forward model in the compressive sensing test was linear using a random measurement matrix.

For this test, we run l_pEKI using only a small ensemble size due to the high computational cost of running the forward model. As in the previous test, we use the multiple batch approach. First, the l_pEKI ensemble of size 50 is initialized around

p	l_1 error	data misfit
2	21.3389	4.1227
1	0.1553	0.5707
0.8	0.0719	0.5682

Table 2: 2D elliptic problem. l_pEKI estimate l_1 error and data misfit for $p = 2, 1$ and 0.8.

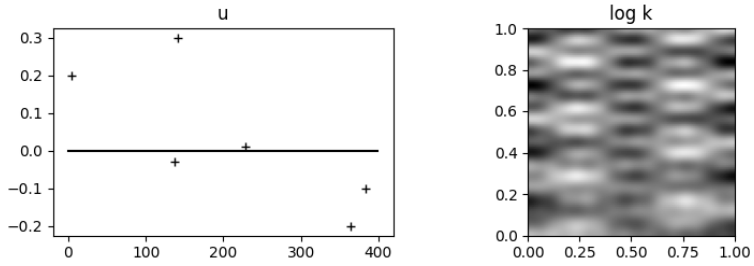
zero with Gaussian perturbations of variance 0.1. After the first five iterations, all components whose magnitudes less than 5×10^{-3} are removed at each iteration. The threshold value is slightly smaller than the smallest magnitude component of the true signal. Over 100 different trials, the average number of nonzero components after 30 iterations is 18 that is smaller than the ensemble size.

The true value of u used in this test and its corresponding log permeability, $\log k$, are shown in the first row of Figure 9 (u is represented as a one-dimensional vector by concatenating the row vectors of $\{u_{ij}\}$). The l_pEKI estimates for $p = 2, 1$, and 0.8 are shown in the second to the fourth rows of Figure 9. Here $p = 0.8$ was the smallest value we can use for l_pEKI due to the numerical overflow in the exponentiation of $\log k$. A smaller p can be used with a smaller variance for ensemble initialization, but the gain is marginal. The results of l_pEKI are similar to the compressive sensing case. $p = 0.8$ has the best performance recovering the four most significant components of u. $p = 1$ has slightly weak magnitudes missing the correct magnitudes of the two most significant components (corresponding to one-dimensional indices 141 and 364). Both cases converge within 20 iterations to yield the best result (see Figure 10 and Table 2 for the time series and numerical values of the l_1 error and data misfit). When $p = 2$, l_pEKI performs the worst; it has the largest l_1 error and data misfit. We note that $p = 2$ uses the result after running 50 iterations at which the estimate converges.

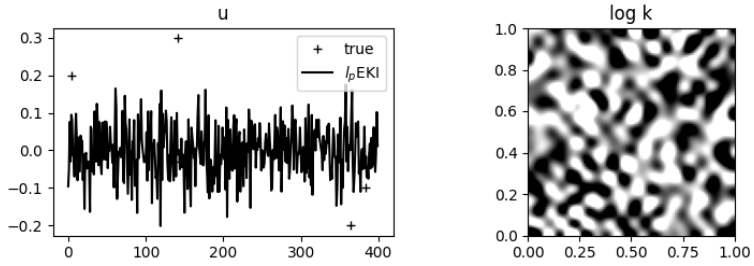
The performance difference between different trials is not significant. The standard deviations of the l_pEKI estimates using 100 trials are shown in Figure 11. The standard deviations for nonzero components are larger than the other components, but the largest standard deviation is less than 3% of the magnitude of the true signal. As in the compressive sensing test, the deviations are slightly smaller for $p < 1$ than $p = 1$.

5. Discussions and conclusions. We have proposed a strategy to implement $l_p, 0 < p \leq 1$, regularization in ensemble Kalman inversion (EKI) to recover sparse structures in the solution of an inverse problem. The l_p-regularized ensemble Kalman inversion (l_pEKI) proposed here uses a transformation to convert the l_p regularization problem to the l_2 regularization problem, which is then solved by the standard EKI with an augmented measurement model used in Tikhonov EKI. We showed a one-to-one correspondence between the local minima of the original and the transformed formulations. Thus a local minimum of the original problem can be obtained by finding a local minimum of the transformed problem. As other iterative methods for non-convex problems, initialization plays a vital role in the proposed method's performance. The effectiveness and robustness of regularized EKI are validated through a suite of numerical tests, showing robust results in recovering sparse solutions using $p \leq 1$.

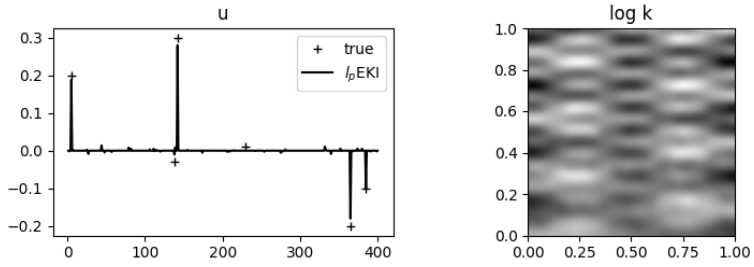
In implementing l_p regularization for EKI, there is a limit on $p < 1$ due to an overflow. One definitive source of the overflow is the transformation ξ that involves



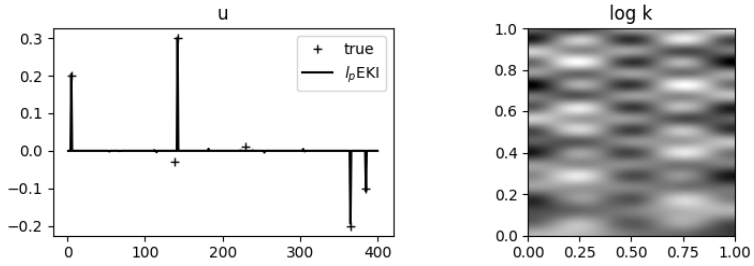
(a) true



(b) $p = 2$



(c) $p = 1$



(d) $p = 0.8$

Fig. 9: 2D elliptic problem. Left column: the true u and l_pEKI estimates for $p = 2, 1$, and 0.8. Right column: $\log k$ of the true and l_pEKI estimates. All plots have the same grey scale. $p = 1$ and 0.8 use the results after 20 iterations while $p = 2$ uses the result after 50 iterations. For the true signal, only the nonzero components are marked.

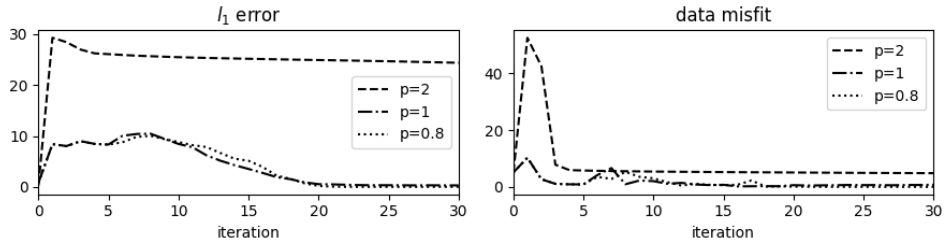


Fig. 10: 2D elliptic problem. l_1 error of the l_pEKI estimates and data misfit.

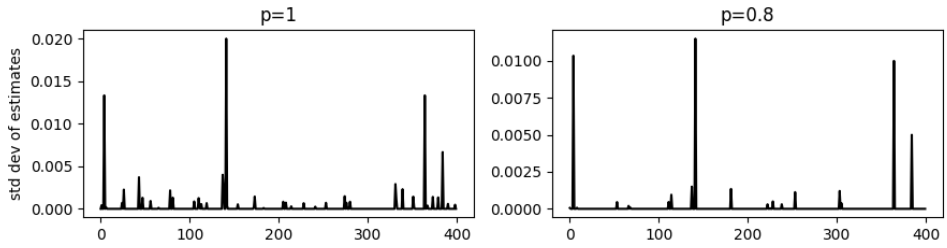


Fig. 11: 2D elliptic problem. Standard deviation of the estimates using 100 trials.

$\frac{2}{p}$ as an exponent. For a small $p < 1$, the transformation ξ can diverge, and thus EKI suffers from instability. One possible workaround is to impose the l_p penalty term directly in the fidelity term instead of transforming it to the l_2 regularization using ξ. The penalty term incorporated in the fidelity term can be achieved by an extended measurement framework similar to Tikhonov EKI but with a nonlinear measurement operator. Also, in the ensemble filters, the filter estimate can diverge to machine infinity under a stringent filter setup, which is called 'catastrophic filter divergence' [19, 17]. It is shown in [22] that one of the mechanisms for the filter instability is related to the measurement operator. As l_p regularization in EKI is implemented through an extended measurement operator, it is natural to investigate a connection between the catastrophic filter divergence and the instability in l_pEKI for $p < 1$. In particular, it is worth considering several methods that prevent the catastrophic filter divergence, including adaptive inflation [33, 24], for stabilizing l_pEKI. The effect of the above-mentioned approaches in stabilizing l_pEKI for $p < 1$ is under investigation and will be reported in another place.

For successful applications of l_pEKI for high-dimensional inverse problems, it is essential to maintain a small ensemble size for efficiency. In the current study, we considered the multiple batch approach. The approach removes non-significant components after each batch, and thus the problem size (i.e., the dimension of the unknown signal) decreases over different batch runs. This approach enabled l_pEKI to use only 50 ensemble members to solve 200 and 400-dimensional inverse problems. Other techniques, such as variance inflation and localization, improve the performance of the standard EKI using a small ensemble size [30]. It would be natural to investigate if these techniques can be extended to l_pEKI to decrease the sampling error of l_pEKI.

In the current study, we have left several variants of l_pEKI for future work.

Weighted l_1 has been shown to recover sparse solutions using fewer measurements than the standard l_1 [7]. It is straightforward to implement weighted l_1 (and further weighted l_p for $p < 1$) in l_pEKI by replacing the identity matrix in (2.10) with another type of covariance matrix corresponding to the desired weights. We plan to study several weighting strategies to improve the performance of l_pEKI. As another variant ofl_pEKI, we plan to investigate the adaptive time-stepping under the continuous limit. The time step for solving the continuous limit equation, which is called 'learning rate' in the machine learning community, is known to affect an optimization solver [11]. The standard ensemble Kaman inversion has been applied to machine learning tasks, such as discovering the vector fields defining a differential equation, using time series data [23] and sparse learning using thresholding [31]. We plan to investigate the effect of an adaptive time-stepping for performance improvements and compare with the sparsity EKI method using thresholding in dimension reduction in machine learning.

REFERENCES

[1] J. L. Anderson, *An ensemble adjustment kalman filter for data assimilation*, Monthly Weather Review, 129 (2001), pp. 2884–2903.

[2] J. M. Bardsley, A. Solonen, H. Haario, and M. Laine, *Randomize-then-optimize: A method for sampling from posterior distributions in nonlinear inverse problems*, SIAM Journal on Scientific Computing, 36 (2014), pp. A1895–A1910.

[3] M. Benning and M. Burger, *Modern regularization methods for inverse problems*, Acta Numerica, 27 (2018), p. 1–111.

[4] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, USA, 2004.

[5] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, 2004.

[6] A. C. Reynolds, M. Zafari, and G. Li, *Iterative forms of the ensemble kalman filter*, (2006).

[7] E. J. Candès, M. B. Wakin, and S. P. Boyd, *Enhancing sparsity by reweighted l_1 minimization*, Journal of Fourier Analysis and Applications, 14 (2008), pp. 877–905.

[8] J. Carrera and S. P. Neuman, *Estimation of aquifer parameters under transient and steady state conditions: 3. application to synthetic and field data*, Water Resources Research, 22 (1986), pp. 228–242.

[9] N. K. Chada, A. M. Stuart, and X. T. Tong, *Tikhonov regularization within ensemble kalman inversion*, SIAM Journal on Numerical Analysis, 58 (2020), pp. 1263–1294.

[10] N. K. Chada and X. T. Tong, *Convergence acceleration of ensemble kalman inversion in nonlinear settings*, (2019). arXiv:1911.02424.

[11] J. Duchi, E. Hazan, and Y. Singer, *Adaptive subgradient methods for online learning and stochastic optimization*, J. Mach. Learn. Res., 12 (2011), p. 2121–2159.

[12] M. Elad, *Sparse and Redundant Representations - From Theory to Applications in Signal and Image Processing.*, Springer, 2010.

[13] G. Evensen, *Data Assimilation: The Ensemble Kalman Filter*, Springer, London, 2009.

[14] R. Fletcher, *Practical methods of optimization*, John Wiley, New York, 2nd ed., 1987.

[15] S. Foucart and H. Rauhut, *A Mathematical Introduction to Compressive Sensing*, Birkhäuser Basel, 2013.

[16] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, The MIT Press, 2016.

[17] G. A. Gottwald and A. J. Majda, *A mechanism for catastrophic filter divergence in data assimilation for sparse observation networks*, onlinear Processes in Geophysics, 20 (2013), pp. 705–712.

[18] M. Hanke, *A regularizing levenberg - marquardt scheme, with applications to inverse groundwater filtration problems*, Inverse Problems, 13 (1997), pp. 79–95.

[19] J. Harlim and A. J. Majda, *Catastrophic filter divergence in filtering nonlinear dissipative systems*, Comm. Math. Sci., 8 (2010), pp. 27–43.

[20] M. Iglesias, K. J. H. Law, and A. Stuart, *Ensemble kalman methods for inverse problems*, Inverse Problems, 29 (2013), p. 045001.

[21] M. A. Iglesias, *A regularizing iterative ensemble kalman method for PDE-constrained inverse problems*, Inverse Problems, 32 (2016), p. 025002.

[22] D. Kelly, A. J. Majda, and X. T. Tong, *Concrete ensemble kalman filters with rigorous catastrophic filter divergence*, Proceedings of the National Academy of Sciences, 112 (2015), pp. 10589–10594, https://doi.org/10.1073/pnas.1511063112, https://www.pnas.org/content/112/34/10589, https://arxiv.org/abs/https://www.pnas.org/content/112/34/10589.full.pdf.

[23] N. B. Kovachki and A. M. Stuart, *Ensemble kalman inversion: a derivative-free technique for machine learning tasks*, Inverse Problems, 35 (2019), p. 095005.

[24] Y. Lee, A. Majda, and D. Qi, *Preventing catastrophic filter divergence using adaptive additive inflation for baroclinic turbulence*, Monthly Weather Review, 145 (2017), pp. 669–682.

[25] A. C. Li, Gaoming; Reynolds, *An iterative ensemble kalman filter for data assimilation*, (2007).

[26] L. V. M.S. Andersen, J. Dahl, *Cvxopt: A python package for convex optimization*. cvxopt.org.

[27] D. Oliver, A. C. Reynolds, and N. Liu, *Inverse Theory for Petroleum Reservoir Characterization and History Matching*, Cambridge University Press, Cambridge, UK, 1st ed., 2008.

[28] B. T. Polyak and A. B. Juditsky, *Acceleration of stochastic approximation by averaging*, SIAM Journal on Control and Optimization, 30 (1992), pp. 838–855.

[29] H. Schaeffer, *Learning partial differential equations via data discovery and sparse optimization*, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 473 (2017), p. 20160446.

[30] C. Schillings and A. M. Stuart, *Analysis of the ensemble kalman filter for inverse problems*, SIAM Journal on Numerical Analysis, 55 (2017), pp. 1264–1290.

[31] T. Schneider, A. M. Stuart, and J.-L. Wu, *Imposing sparsity within ensemble kalman inversion*, (2020). arXiv:2007.06175.

[32] Z. Wang, J. M. Bardsley, A. Solonen, T. Cui, and Y. M. Marzouk, *Bayesian inverse problems with l_1 priors: A randomize-then-optimize approach*, SIAM Journal on Scientific Computing, 39 (2017), pp. S140–S166.

[33] A. J. M. X. T. Tong and D. Kelly, *Nonlinear stability of the ensemble kalman filter with adaptive covariance inflation*, Comm. Math. Sci, 14 (2016), pp. 1283–1313.