Ensemble Kalman Methods With Constraints

David J. Albers[1,2], **Paul-Adrien Blancquart**[3], **Matthew E. Levine**[4], **Elnaz Esmaeilzadeh Seylabi**[5], **Andrew Stuart**[4]

[1] Department of Biomedical Informatics, Columbia University, New York, NY 10032

[2] Department of Pediatrics, Division of Informatics, University of Colorado Medicine, Aurora, CO 80045

[3] Mines ParisTech, PSL Research University, Paris, France

[4] Department of Computational and Mathematical Sciences, California Institute of Technology, Pasadena, CA 91125

[5] Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125

E-mail: `david.albers@ucdenver.edu`, `paul-adrien.blancquart@mines-paristech.fr`, `mlevine@caltech.edu`, `elnaz@caltech.edu`, `astuart@caltech.edu`

Abstract. Ensemble Kalman methods constitute an increasingly important tool in both state and parameter estimation problems. Their popularity stems from the derivative-free nature of the methodology which may be readily applied when computer code is available for the underlying state-space dynamics (for state estimation) or for the parameter-to-observable map (for parameter estimation). There are many applications in which it is desirable to enforce prior information in the form of equality or inequality constraints on the state or parameter. This paper establishes a general framework for doing so, describing a widely applicable methodology, a theory which justifies the methodology, and a set of numerical experiments exemplifying it.

Keywords: *ensemble Kalman methods, equality and inequality constraints, derivative-free optimization, convex optimization*

Submitted to: *Inverse Problems*

1. Introduction

1.1. Overview

Kalman filter based methods have been enormously successful in both state and parameter estimation problems. However, a major disadvantage of such methods is that they do not naturally take constraints into account. The ability to constrain a system often has a number of advantages that can play an important role in state and parameter estimation: they can be used to enforce physicality of modeled systems (non-negativity of physical quantities, for example); relatedly they can be used to ensure that computational models are employed only within state and parameter regimes where the model is well-posed; and finally the application of constraints may provide robustness to outlier data. Resulting improvements in algorithmic efficiency and performance, by means of enforcing constraints, has been demonstrated in the recent literature in a diverse set of fields, including process control [1], biomechanics [2], cell energy metabolism [3], medical imaging [4], engine health estimation [5], weather forecasting [6], chemical engineering [7], and hydrology [8].

In the probabilistic view of filtering methods, constraints may be introduced by moving beyond the Gaussian assumptions that underpin Kalman methods and imposing constraints through the prior distributions on states and/or parameters. This, however, can create significant computational burden as the resulting distributions cannot be represented in closed form, through a finite number of parameters, in the way that Gaussian distributions can be. In this paper, we circumvent this issue by taking the viewpoint that ensemble Kalman methods constitute a form of derivative-free optimization methodology, eschewing the probabilistic interpretation. The ensemble is used to calculate surrogates for derivatives. With this optimization perspective, constraints may be included in a natural way. Standard ensemble Kalman methods employ a quadratic optimization problem encapsulating relative strengths of belief in the predictions of the model and the data; these optimization problems have explicit analytic solutions. To impose constraints the optimization problem is solved only within the constraint set; when the constraints form a non-empty closed convex set, this constrained optimization problem has a unique solution.

In this introductory section, we give a literature review describing existing work in this setting, we describe the contributions in this paper, and we outline notation used throughout.

1.2. Literature Review

Overviews of state estimation using Kalman based methods may be found in [9, 10, 11, 12]. The focus of this article is on ensemble based Kalman methods, introduced by Evensen in [13] and further developed in [14, 9]. The extension of the ensemble Kalman methodology to parameter estimation and inverse problems is overviewed in [15], especially for oil reservoir applications, and in an application-neutral formulation in [16]. Equipping Kalman-based methods with constraints can be desirable for a variety of inter-linked reasons described in the previous subsection: to enforce known physical boundaries in order to improve estimation accuracy; to operationalize filtering of a model which is ill-posed in subsets of its state or parameter space; and to provide robustness to noisy data and outlier events.

In extending the Kalman filter to non-Gaussian settings, a number of methods may be considered. Particle filters provide the natural methodology if propagation of probability distributions is required for state [17] or parameter [18] estimation. In the optimization setting, there are three primary methodologies: the extended Kalman filter, the unscented Kalman filter and the ensemble Kalman filter. The extended Kalman filter is based on linearization of the nonlinear system and therefore needs the computation of derivatives for propagation of the state covariance; this makes them unattractive in high dimensional problems. Unscented and ensemble Kalman filters, on the other hand, can be considered as particle-based methods which are derivative-free. In the unscented Kalman filter, the particles (sigma points) are chosen deterministically and are propagated through the nonlinear system to approximate the covariance, which is then corrected using the Kalman gain to compute the new sigma points. In the ensemble Kalman filter, the particles (ensemble members) are chosen randomly from the initial ensemble and are propagated through the dynamical system and corrected using the Kalman gain without needing to maintain the covariance.

In [19], and more recently in [20], overviews of different ways to impose constraints in linear and nonlinear state estimation are presented. To ensure that the estimates satisfy the constraints, moving horizon based estimators that solve a constrained optimization problem have been proposed [21, 22]. The paper [23] proposed a recursive nonlinear dynamic data reconciliation (RNDDR) approach based on extended Kalman filtering to ensure that state and parameter estimates satisfy the imposed bounds and constraints. The updated state estimates in this method are obtained by solving an optimization problem instead of using the Kalman gain. The resulting covariance calculations are, however, still similar to the Kalman filter: that is, unconstrained propagation and correction involving the Kalman gain, which can affect the accuracy of the estimates. To eliminate this deficiency, [24] proposed a Kullback-Leibler based method to update states and error covariances by solving a convex optimization problem involving conic constraints.

On the other hand, the paper [25] combined the concept of the unscented transformation [26] with the RNDDR formulation. In the prediction step, they propose step sizes to scale sigma points asymmetrically to better approximate the covariance information in the presence of lower and upper bounds. Then, for the update of each sigma point, they solve a constrained optimization problem. One disadvantage of this procedure is that the chosen step sizes for scaling the sigma points can only ensure the bound constraints. The paper [1] also tested various algorithms based on constrained optimization, projection [27] and truncation [5] to enforce bound constraints on unscented Kalman filtering. The paper [28] developed a class of estimators named constrained unscented recursive estimators to address the limitations of the unscented RNDDR method using optimization-based projection algorithms for obtaining sigma points in the presence of convex, non-convex and bound constraints.

As mentioned earlier, since the corrected covariance is used to compute the sigma points, unscented formulations always require enforcing constraints in both propagation and correction/update steps. In contrast, ensemble-based methods only require constraints to be enforced in the update step. In this context, the paper [8] tested projection and accept/reject methods to constrain ensemble members in a post-processing step, after application of the unconstrained ensemble Kalman filter. In the former, they project the updated ensemble members to the feasible space if they violate the constraints and in the latter they enforce the updated ensemble members to obey the constraints by resampling the dynamic and/or data model errors. On the other hand, [29, 30] proposed updating the state estimates in ensemble Kalman filtering by solving a constrained optimization problem while truncating the Gaussian distribution of the initial ensemble. The paper [6] demonstrated how to enforce a physics-based conservation law on an ensemble Kalman filtering based state estimation problem by formulating the filter update as a set of quadratic programming problems arising from a linear data acquisition model subject to linear constraints. Here we develop this body of work on constraining ensemble Kalman techniques, providing a unifying framework with an underpinning theoretical basis.

1.3. Our Contribution

The preceding literature review demonstrates that the imposition of constraints on state and parameter estimation procedures is highly desirable. It also indicates that ensemble Kalman methods offer the most natural context in which to attempt to do this, as extended Kalman methods do not scale well to high dimensional state or parameter space, whilst the unscented filter does not lend itself as naturally to the incorporation of constraints.

In this paper we build on the application-specific papers [8, 6] which demonstrate how to impose some specific constraints on ensemble based parameter and state estimation problems respectively. We formulate a very general methodology which is application-neutral and widely applicable, thereby making the ideas in [8, 6] accessible to a wide community of researchers working in inverse problems and state estimation. We also describe a straightforward mathematical analysis which demonstrates that the resulting algorithms are well-defined since they involve the solution of quadratic minimization problems subject to convex constraints at each step of the algorithm; these optimization problems have a unique solution. And finally we showcase the methodology on two applications, one from biomedicine and one from seismology.

Section 2 outlines the ensemble Kalman (EnKF) methodology for state estimation, with and without constraints. In section 3 the same program is carried out for ensemble Kalman inversion (EKI). Section 4 describes the numerical experiments which illustrate the foregoing ideas.

1.4. Notation

Throughout the paper we use $\mathbb{N}$ to denote the positive integers $\{1, 2, 3, \cdots\}$ and $\mathbb{Z}^+$ to denote the non-negative integers $\mathbb{N} \cup \{0\} = \{0, 1, 2, 3, \cdots\}$. The matrix I_M denotes the identity on $\mathbb{R}^M$. We use $|\cdot|$ to denote the Euclidean norm, and the corresponding inner-product is denoted $\langle \cdot, \cdot \rangle$. A symmetric, square matrix A is positive definite (resp. positive semi-definite) if the quadratic form $\langle u, Au \rangle$ is positive (resp. non-negative) for all $u \neq 0$. By $|\cdot|_B$ we denote the weighted norm defined by $|v|_B^2 = v^* B^{-1} v$ for any positive-definite B. The corresponding weighted Euclidean inner-product is given by $\langle \cdot, \cdot \rangle_B := \langle \cdot, B^{-1} \cdot \rangle$. We use $\otimes$ to denote the outer product between two vectors: $(a \otimes b)c = \langle b, c \rangle a$.

2. Ensemble Kalman State Estimation

2.1. Filtering Problem

Consider the discrete-time dynamical system with noisy state transitions and noisy observations in the form:

$$\begin{aligned}
\text{Dynamics Model:}\quad & v_{j+1} = \Psi(v_j) + \xi_j, \quad j \in \mathbb{Z}^+ \\
\text{Data Model:}\quad & y_{j+1} = Hv_{j+1} + \eta_{j+1}, \quad j \in \mathbb{Z}^+ \\
\text{Probabilistic Structure:}\quad & v_0 \sim N(m_0, C_0), \quad \xi_j \sim N(0, \Sigma), \quad \eta_j \sim N(0, \Gamma) \\
\text{Probabilistic Structure:}\quad & v_0 \perp \{\xi_j\} \perp \{\eta_j\} \text{ independent}
\end{aligned}$$

We assume that $\mathcal{H}_1, \mathcal{H}_2$ are separable Hilbert spaces. Then $v_j \in \mathcal{H}_1$, and $\Psi : \mathcal{H}_1 \mapsto \mathcal{H}_1$ is the state-transition operator. The operator $H : \mathcal{H}_1 \mapsto \mathcal{H}_2$ is the linear observation operator and $y_j \in \mathcal{H}_2$. The covariance operators C_0, Σ are assumed trace-class on $\mathcal{H}_1$, and Γ on $\mathcal{H}_2$ which ensures that the initial condition v_1 and the noises ξ_j and η_j live in $\mathcal{H}_1, \mathcal{H}_1$ and $\mathcal{H}_2$ (respectively) with probability one. The objective of filtering is to estimate the state v_j of the dynamical systems at time j, given the data $\{y_\ell\}_{\ell=1}^{j}$. Throughout this paper we derive our theoretical results in the setting where $\mathcal{H}_1$ and $\mathcal{H}_2$ are finite dimensional; however the update formulae we derive are well-defined in the general Hilbert space setting and this fact is important because it means that the methods derived have a robustness to mesh refinement and similar procedures arising when the problem of interest is specified via a partial differential equation, or other infinite dimensional problem.

Remark 2.1. We restrict attention to linear observation operators H because this leads to solvable quadratic optimization problems within the context of Kalman-based methods. In principle, a non-linear observation operator could be used, but the optimization problems defining the algorithms arising in this paper might not have a unique solution in this setting.

2.2. Ensemble Kalman Filter

The ensemble Kalman filter is a particle-based sequential optimization approach to the state estimation problem. The particles are denoted by $\{v_j^{(n)}\}_{n=1}^N$ and represent a collection of N candidate state estimates at time j. The method proceeds as follows. The state of all the particles at time $j+1$ are predicted using the dynamics model to give $\{\widehat{v}_{j+1}^{(n)}\}_{n=1}^N$. The resulting empirical covariance of the particles is then used to define the objective function $I_{\text{filter},j,n}(v)$, which encapsulates the model-data compromise. This is minimized in order to obtain the updates $\{v_{j+1}^{(n)}\}_{n=1}^N$.

The prediction step is

$$\widehat{v}_{j+1}^{(n)} = \Psi(v_j^{(n)}) + \xi_j^{(n)}, n = 1, ..., N \tag{1a}$$

$$\widehat{m}_{j+1} = \frac{1}{N} \sum_{n=1}^{N} \widehat{v}_{j+1}^{(n)} \tag{1b}$$

$$\widehat{C}_{j+1} = \frac{1}{N} \sum_{n=1}^{N} \Big(\widehat{v}_{j+1}^{(n)} - \widehat{m}_{j+1}\Big)\Big(\widehat{v}_{j+1}^{(n)} - \widehat{m}_{j+1}\Big)^T. \tag{1c}$$

Here we have $\xi_j^{(n)} \sim N(0, \Sigma)$ i.i.d.. Because the empirical covariance contains only $N-1$ independent pieces of information, (1c) is sometimes scaled by $N-1$ and not N; making this change would lead to no changes in the statements and proofs of all the theorems, and would only affect the definition of covariance within the algorithms.

Let $\mathcal{R}(\widehat{C}_{j+1})$ denote the range of $\widehat{C}_{j+1}$. The update step is then

$$v_{j+1}^{(n)} = \underset{v}{\operatorname{argmin}}\, I_{\text{filter,j,n}}(v) \tag{2}$$

where

$$I_{\text{filter},j,n}(v) := \begin{cases} \frac{1}{2} \mid y_{j+1}^{(n)} - Hv \mid_{\Gamma}^2 + \frac{1}{2} \mid v - \widehat{v}_{j+1}^{(n)} \mid_{\widehat{C}_{j+1}}^2 & \text{if } v - \widehat{v}_{j+1}^{(n)} \in \mathcal{R}(\widehat{C}_{j+1}). \\ \infty & \text{otherwise.} \end{cases} \tag{3}$$

It can be useful to rewrite the objective function for the optimization problem in an equivalent and more standard form for input to software:

$$\begin{cases} \frac{1}{2} v^T \Big(H^T \Gamma^{-1} H + \widehat{C}_{j+1}^{-1}\Big) v - \Big(\widehat{C}_{j+1}^{-1^T} \widehat{v}_{j+1}^{(n)} + H^T \Gamma^{-1^T} y_{j+1}^{(n)}\Big)^T v & \text{if } v - \widehat{v}_{j+1}^{(n)} \in \mathcal{R}(\widehat{C}_{j+1}). \\ \infty & \text{otherwise.} \end{cases}$$

The $y_{j+1}^{(n)}$ are either identical to the data y_{j+1}, or found by perturbing it randomly.

Note that $\widehat{C}_{j+1}$ is an operator of rank at most $N-1$, and thus can only be invertible when $N-1$ is larger than the dimension of $\mathcal{H}_1$. For moderate- and high-dimensional systems, it is often impractical to satisfy this condition. However, the minimizing solution can be found by regularizing $\widehat{C}_{j+1}$ by addition of ϵI for $\epsilon > 0$, deriving the update equations and then letting $\epsilon \to 0$. We give the resulting formulae, and then justify them immediately afterwards, in the following subsubsection. Alternatively it is possible to directly seek a solution in $\mathcal{R}(\widehat{C}_{j+1})$, which is a subspace of dimension $N-1$; this is done in the subsequent subsubsection.

2.2.1. Formulation In The Original Variables The well-known Kalman update formulae arising from solution of the minimization problem (3) are as follows:

$$S_{j+1} = H\widehat{C}_{j+1}H^T + \Gamma \tag{4a}$$

$$K_{j+1} = \widehat{C}_{j+1}H^T S_{j+1}^{-1} \qquad \text{(Kalman Gain)} \tag{4b}$$

$$y_{j+1}^{(n)} = y_{j+1} + s\eta_{j+1}^{(n)}, n = 1, ..., N \tag{4c}$$

$$v_{j+1}^{(n)} = (I - K_{j+1}H)\widehat{v}_{j+1}^{(n)} + K_{j+1}y_{j+1}^{(n)}, n = 1, ..., N \tag{4d}$$

Here $\eta_j^{(n)} \sim N(0,\Gamma)$ i.i.d. and the constant s takes value 0 or 1. When $s = 1$ the $y_{j+1}^{(n)}$ are referred to as perturbed observations. The choice $s = 1$ is made to ensure the correct statistics of the updates in the linear Gaussian setting when a probabilistic viewpoint is taken, and more generally to introduce diversity into the ensemble procedure when an optimization viewpoint is taken. Derivation of the formulae may be found in [31]. In brief the formulae arise from completing the square in the objective function $I_{\text{filter},j,n}(\cdot)$ and then applying the Sherman–Morrison formula to rewrite the updates in the data space rather than state space; the latter is advantageous in many applications where $\mathcal{H}_2$ has dimension much smaller than $\mathcal{H}_1$.

We summarize with the following pseudo-code:

Algorithm 1 EnKF Algorithm

```
1: Choose $\{v_0^{(n)}\}_{n=1}^N$, $j = 0$
2: Predict $\{\widehat{v}_{j+1}^{(n)}\}_{n=1}^N$, $\widehat{C}_{j+1}$ from (1)
3: Update $\{v_{j+1}^{(n)}\}_{n=1}^N$ from (4)
4: $j \leftarrow j + 1$, go to 2.
```

An equivalent formulation of the minimization problem is now given by means of a penalized Lagrangian approach to incorporate the property that the solution of the optimization problem lies in the range of the empirical covariance. The perspective is particularly useful when further constraints are imposed on the solution of the optimization problem.

Theorem 2.2. *Suppose that the dimensions of $\mathcal{H}_1$ and $\mathcal{H}_2$ are finite. Let j be in $\mathbb{Z}^+$ and $1 \leq n \leq N$. Define $y' = y_{j+1}^{(n)} - H\widehat{v}_{j+1}^{(n)}$. Then the update formulae (1), (4) may be given alternatively by*

$$v_{j+1}^{(n)} = \widehat{v}_{j+1}^{(n)} + \operatorname*{argmin}_{(a,v')\in\mathcal{A}} \left(\frac{1}{2} \mid y' - Hv' \mid_\Gamma^2 + \frac{1}{2}\langle a, v'\rangle\right) \tag{5}$$

where $\mathcal{A} = \{(a, v') \in \mathcal{H}_1 \times \mathcal{H}_1 : \widehat{C}_{j+1} a = v'\}$ *and the argmin is projected from the pair* (a, v') *onto the* v' *coordinate only. Moreover* $v_{j+1}^{(n)} = \lim_{\epsilon \to 0} v_\epsilon$ *with*

$$v_\epsilon = \operatorname*{argmin}_{v \in \mathcal{H}_1} \left(\frac{1}{2} \mid y_{j+1}^{(n)} - Hv \mid_\Gamma^2 + \frac{1}{2} \mid \widehat{v}_{j+1}^{(n)} - v \mid_{\widehat{C}_\epsilon}^2 \right)$$

and $\widehat{C}_\epsilon = \widehat{C}_{j+1} + \epsilon I$*.*

Proof. For notational convenience denote $\widehat{C} = \widehat{C}_{j+1}$ and see that the minimization (5) is performed under the constraint $\widehat{C}a = v'$. Then notice that $\langle a, v' \rangle = \mid v' \mid_{\widehat{C}}^2$ with v' lying in the range of the operator $\widehat{C}$; this is a convex constraint. The restriction of $\widehat{C}$ over the constraint set is positive definite which means that the quadratic objective function, now depending only on v', is strongly convex. Therefore the problem has a unique solution and its Lagrangian is written as:

$$\mathcal{L}(v', a, \lambda) = \frac{1}{2} |y' - Hv'|_\Gamma^2 + \frac{1}{2} \langle a, v' \rangle + \langle \lambda, \widehat{C}a - v' \rangle$$

To express optimality conditions compute the derivatives and set them to zero:

$$\begin{aligned} -H^T \Gamma^{-1}(y' - Hv') + \frac{1}{2}a - \lambda &= 0, \\ \frac{1}{2}v' + \widehat{C}\lambda &= 0, \\ v' - \widehat{C}a &= 0. \end{aligned}$$

The last two equations imply that $\widehat{C}(2\lambda + a) = 0$. Thus we set $\lambda = -\frac{1}{2}a$ and drop the second equation, replacing the first by

$$-H^T \Gamma^{-1}(y' - H\widehat{C}a) + a = 0.$$

Solving this for a gives

$$\begin{aligned} v_{j+1}^{(n)} &= \widehat{v}_{j+1}^{(n)} + v' \\ &= \widehat{v}_{j+1}^{(n)} + \widehat{C}a \\ &= \widehat{v}_{j+1}^{(n)} + \widehat{C}(H^T \Gamma^{-1} H \widehat{C} + I)^{-1} H^T \Gamma^{-1} y' \\ &= \widehat{v}_{j+1}^{(n)} + \widehat{C}(H^T \Gamma^{-1} H \widehat{C} + I)^{-1} H^T \Gamma^{-1} (y_{j+1}^{(n)} - H\widehat{v}_{j+1}^{(n)}) \\ &= (I - K_{j+1}H)\widehat{v}_{j+1}^{(n)} + K_{j+1} y_{j+1}^{(n)} \,. \end{aligned}$$

It remains to show that K_{j+1} agrees with the prescription given in the formulae above. To see this we note that if we choose S to be any matrix satisfying $K_{j+1} = \widehat{C}H^T S^{-1}$ then

$$H^T S^{-1} = (H^T \Gamma^{-1} H \widehat{C} + I)^{-1} H^T \Gamma^{-1}$$

so that

$$(H^T\Gamma^{-1}H\widehat{C}+I)H^T = H^T\Gamma^{-1}S.$$

Thus

$$H^T\Gamma^{-1}H\widehat{C}H^T + H^T = H^T\Gamma^{-1}S$$

which may be achieved by choosing any S so that

$$\Gamma^{-1}(H\widehat{C}H^T+\Gamma) = \Gamma^{-1}S$$

and multiplication by Γ gives the desired formula for S_{j+1}.

Concerning the alternative representation of the solution, we note that $H^T\Gamma^{-1}H + \widehat{C}_\epsilon^{-1}$ is strictly positive definite and hence the related quadratic function is strongly convex. As a consequence we have existence and uniqueness of the solution, and the optimality condition becomes,

$$(H^T\Gamma^{-1}H + \widehat{C}_\epsilon^{-1})v_\epsilon = H^T\Gamma^{-1}y_{j+1}^{(n)} + \widehat{C}_\epsilon^{-1}\widehat{v}_{j+1}^{(n)}\,.$$

Then if we apply Woodbury matrix identity we obtain

$$v_\epsilon = (\widehat{C}_\epsilon - \widehat{C}_\epsilon H^T(H\widehat{C}_\epsilon H^T+\Gamma)^{-1}H\widehat{C}_\epsilon)(H^T\Gamma^{-1}y_{j+1}^{(n)} + \widehat{C}_\epsilon^{-1}\widehat{v}_{j+1}^{(n)})$$

and rearranging the terms:

$$v_\epsilon = (I - \widehat{C}_\epsilon H^T(H\widehat{C}_\epsilon H^T+\Gamma)^{-1}H)\widehat{v}_{j+1}^{(n)} + \widehat{C}_\epsilon H^T(H\widehat{C}_\epsilon H^T+\Gamma)^{-1}y_{j+1}^{(n)}.$$

Finally, as $A \mapsto A^{-1}$ is continuous over the set of invertible matrices, letting $\epsilon \to 0$ gives:

$$\lim_{\epsilon\to 0} v_\epsilon = (I-K_{j+1}H)\widehat{v}_{j+1}^{(n)} + K_{j+1}y_{j+1}^{(n)}$$

which concludes the proof. □

2.2.2. Formulation In Range Of The Covariance The form of the minimization problem for each individual particle has a special structure which follows from the fact that the predicted covariance is computed empirically and is a sum of rank one matrices. This allows us to seek the solution of the minimization problem as a linear combination of a given set of vectors, and to minimize over the scalars which define this linear combination. This reformulation of the optimization problem is useful if the number of ensemble members N is much smaller than the dimension of the data space, where the inversion of S takes place to form Kalman gain K.

In order to implement the minimization in the N dimensional subspace we note that $I_{\text{filter},j,n}(v)$ is infinite unless

$$v - \widehat{v}_{j+1}^{(n)} = \widehat{C}_{j+1} a$$

for some $a \in \mathbb{R}^n$. From the structure of $\widehat{C}_{j+1}$ given in (1c) it follows that

$$v = \widehat{v}_{j+1}^{(n)} + \frac{1}{N} \sum_{m=1}^{N} b_m e^{(m)}, \quad e^{(m)} := \widehat{v}_{j+1}^{(m)} - \widehat{m}_{j+1}. \tag{7}$$

Here each unknown parameter $b_m \in \mathbb{R}$ and $b := \{b_m\}_{m=1}^{N}$, is the unknown vector to be determined. This form for v follows from the fact that

$$\widehat{C}_{j+1} = \frac{1}{N} \sum_{m=1}^{N} e^{(m)} \otimes e^{(m)} \tag{8}$$

which in turn implies that

$$\widehat{C}_{j+1} a = \frac{1}{N} \sum_{m=1}^{N} b_m e^{(m)}. \tag{9}$$

Note that the unknown vector b depends on n as we need to solve the constrained minimization problem for each of the particles, indexed by $n = 1, \dots, N$; we have suppressed the dependence of b on n for notational simplicity.

The expression (7) for v in terms of the $e^{(m)}$ can be substituted into (3) to obtain a functional $J_{\text{filter},j,n}(b)$ to be minimized over $b \in \mathbb{R}^N$, because v is an affine function of b. Equation (7) may be written in compact form as

$$v = \widehat{v}_{j+1}^{(n)} + Bb \tag{10}$$

where B is the linear mapping from $\mathbb{R}^N$ into $\mathcal{H}_1$ defined by

$$Bb := \frac{1}{N} \sum_{m=1}^{N} b_m e^{(m)}.$$

We now identify $J_{\text{filter},j,n}(b)$. We note that (9) is solved by taking

$$b_m = \langle e^{(m)}, a \rangle.$$

Now note that

$$\frac{1}{2} \mid v - \widehat{v}_{j+1}^{(n)} \mid_{\widehat{C}_{j+1}}^{2} = \frac{1}{2} \langle a, \widehat{C}_{j+1} a \rangle = \frac{1}{2N} \sum_{m=1}^{N} b_m^2.$$

Using this and (10) in the definition of $I_{\text{filter},j,n}(v)$ we obtain

$$J_{\text{filter},j,n}(b) = I_{\text{filter},j,n}\Big(\widehat{v}_{j+1}^{(n)} + Bb\Big)$$

and hence, from (3),

$$\begin{aligned} J_{\text{filter},j,n}(b) :=& \frac{1}{2} \mid y_{j+1}^{(n)} - H\widehat{v}_{j+1}^{(n)} - HBb \mid_{\Gamma}^{2} + \frac{1}{2N}|b|^2 && \text{(11a)} \\ =& \frac{1}{2} b^T\Big(B^T H^T \Gamma^{-1} HB + \frac{1}{N} I\Big)b - \Big(B^T H^T \Gamma^{-1}(y_{j+1}^{(n)} - H\widehat{v}_{j+1}^{(n)})\Big)^T b + \text{const.} && \text{(11b)} \end{aligned}$$

Once b is determined it may be substituted back into (10) to obtain the solution to the minimization problem.

The preceding considerations also yield the following result, concerning the unconstrained Kalman minimization problem; its proof is a corollary of the more general Theorem 2.4 from the next subsection, which includes constraints in the minimization problem.

Corollary 2.3. *Suppose that the dimensions of $\mathcal{H}_1$ and $\mathcal{H}_2$ are finite. Given the prediction (1a), the unconstrained Kalman update formulae may be found by minimizing $J_{\text{filter},j,n}(b)$ from (11) with respect to b and substituting into (10).*

We summarize the ensemble Kalman state estimation algorithm, using minimization over the vector b, in the following pseudo-code:

Algorithm 2 EnKF Algorithm formulated in range of covariance

```
1: Choose $\{v_0^{(n)}\}_{n=1}^{N}$, $j = 0$
2: Predict $\{\widehat{v}_{j+1}^{(n)}, e^{(n)}\}_{n=1}^{N}$, from (1)
3: Optimize $\{b^{(n)}\}_{n=1}^{N}$ as argmin of (11)
4: Update $v_{j+1}^{(n)} = \widehat{v}_{j+1}^{(n)} + Bb^{(n)}$ from (10)
5: $j \leftarrow j + 1$, go to 2.
```

2.3. Constrained Ensemble Kalman Filter

In this subsection we introduce linear equality and inequality constraints on the state variable into the ensemble Kalman filter. We make prediction according to

(1), and then incorporate data by solving the minimization problem (3) subject to the additional constraints

$$Fv = f, \tag{12a}$$

$$Gv \preceq g. \tag{12b}$$

Here F and G are linear mappings which, respectively, take the state v into the number of equality and inequality constraints; the notation $\preceq$ denotes inequality componentwise.

2.3.1. Formulation In The Original Variables The preceding considerations lead to the following algorithm for ensemble Kalman filtering subject to constraints (the theoretical justification for using this algorithm follows from Theorem 2.4 below):

Algorithm 3 Constrained EnKF Algorithm

```
Choose $\{v_0^{(n)}\}_{n=1}^N$, $j = 0$
Predict $\{\widehat{v}_{j+1}^{(n)}\}_{n=1}^N$, $\widehat{C}_{j+1}$ from (1)
Update $\{v_{j+1}^{(n)}\}_{n=1}^N$ from (4)
for $n = 1 : N$
    if $v_{j+1}^{(n)}$ violates constraints in (12)
        $v_{j+1}^{(n)} \leftarrow$ argmin of (3) subject to (12)
    end if
end for
$j \leftarrow j + 1$, go to 2.
```

2.3.2. Formulation In Range Of The Covariance The linear constraints (12) can be rewritten in terms of the vector b, by means of (10), as follows:

$$FBb = f - F\widehat{v}_{j+1}^{(n)}, \tag{13a}$$

$$GBb \preceq g - G\widehat{v}_{j+1}^{(n)}. \tag{13b}$$

We may thus predict and then optimize the objective function $J_{\text{filter},j,n}(b)$, given by (11), subject to the constraints (13). Implementation of this leads to following algorithm for ensemble Kalman filtering subject to constraints:

Algorithm 4 Constrained EnKF Algorithm formulated in range of covariance

```
1: Choose $\{v_0^{(n)}\}_{n=1}^N$, $j = 0$
2: Predict $\{\widehat{v}_{j+1}^{(n)}, e^{(n)}\}_{n=1}^N$, from (1)
3: Update $b^{(n)} \leftarrow$ argmin of (11), $v_{j+1}^{(n)} = \widehat{v}_{j+1}^{(n)} + Bb^{(n)}$ from (10)
4: for $n = 1 : N$
5:     if $v_{j+1}^{(n)}$ violates constraints in (12)
6:         $b^{(n)} \leftarrow$ argmin of (11) subject to (13)
7:         Update $v_{j+1}^{(n)} = \widehat{v}_{j+1}^{(n)} + Bb^{(n)}$ from (10)
8:     end if
9: end for
10: $j \leftarrow j + 1$, go to 2.
```

Justification for the use of this algorithm, working in the constrained space parameterized by b, is a consequence of the following:

Theorem 2.4. *Suppose that the dimensions of $\mathcal{H}_1$ and $\mathcal{H}_2$ are finite. The problem of finding $v_{j+1}^{(n)}$ as the minimizer of $I_{\text{filter},j,n}(v)$ subject to the constraints (12) is equivalent to finding b to minimize $J_{\text{filter},j,n}(b)$ subject to the constraints (13) and then using (10) to find $v_{j+1}^{(n)}$ from b. Furthermore, both of these constrained minimization problems have a unique solution provided that the constraint sets are non-empty.*

Proof. For notational convenience set $\widehat{v} = \widehat{v}_{j+1}^{(n)}$, $y = y_{j+1}^{(n)}$, $y' = y - H\widehat{v}$, $\widehat{C} = \widehat{C}_{j+1}$ and $\widehat{C}_\epsilon = \widehat{C}_{j+1} + \epsilon I$.

Denote

$$\begin{aligned} v^* = &\underset{v'}{\text{argmin}} && \frac{1}{2} \mid y' - Hv' \mid_\Gamma^2 + \frac{1}{2}\langle a, v'\rangle \\ &\text{subject to} && \bullet\, \widehat{C}a = v' \\ &&& \bullet\, Fv' = f - F\widehat{v} \\ &&& \bullet\, Gv' \preceq g - G\widehat{v} \end{aligned} \tag{14}$$

$$\begin{aligned} v_\epsilon = &\underset{v}{\text{argmin}} && \frac{1}{2} \mid y - Hv \mid_\Gamma^2 + \frac{1}{2} \mid v - \widehat{v} \mid_{\widehat{C}_\epsilon}^2 \\ &\text{subject to} && \bullet\, Fv = f \\ &&& \bullet\, Gv \preceq g \end{aligned} \tag{15}$$

and

$$
\begin{aligned}
J(v) &= \frac{1}{2} \mid y - Hv \mid_{\Gamma}^{2} + \frac{1}{2} \mid v - \widehat{v} \mid_{\widehat{C}}^{2} \\
J_{\epsilon}(v) &= \frac{1}{2} \mid y - Hv \mid_{\Gamma}^{2} + \frac{1}{2} \mid v - \widehat{v} \mid_{\widehat{C}_{\epsilon}}^{2}
\end{aligned}
$$

The part of the statement of Theorem 2.4 concerning existence of a minimizer is a consequence of the Lemma 2.5 stated and proved below. The second part, concerning the equivalence of minimization over b and over v (or v') was shown prior to the theorem statement. This concludes the proof. □

Lemma 2.5. *Suppose that the constraint sets of* (14) *and* (15) *are non empty, then v^* exists and is unique and for all $\epsilon > 0$, v_ϵ exists and is unique. Furthermore $\underset{\epsilon \to 0}{lim}\, v_\epsilon = v^* + \widehat{v}$.*

Proof. To prove existence and uniqueness of the solution of (14), notice that it can be reformulated as

$$
\begin{array}{ll}
\underset{v'}{\operatorname{argmin}} & J(v' + \widehat{v}) \\
\text{subject to} & \bullet\, \widehat{C}a = v' \\
 & \bullet\, Fv' = f - F\widehat{v} \\
 & \bullet\, Gv' \preceq g - G\widehat{v}
\end{array}
$$

and that the restriction of $\widehat{C}$ over its range is strictly positive definite. Hence J is a strongly convex function being minimized over a non empty closed convex set. From standard theory v^* exists and is unique. Then as $\widehat{C}_\epsilon$ is strictly positive definite, the same type of arguments provide existence and uniqueness of v_ϵ.

Now we prove the second part of the lemma. We note that $v^* + \widehat{v}$ matches the constraints of (15). It follows that for all $\epsilon > 0$, $J_\epsilon(v_\epsilon) \leq J_\epsilon(v^* + \widehat{v})$. Then let us prove that $J_\epsilon(v^* + \widehat{v}) \underset{\epsilon \to 0}{\to} J(v^* + \widehat{v})$. First denote by $\lambda_1 \leq \cdots \leq \lambda_{N-1}$ the strictly positive eigenvalues of $\widehat{C}$ (recall that $\widehat{C}$ is symmetric positive semidefinite and that $\operatorname{rank}(\widehat{C}) = N - 1$ almost surely). Hence $\widehat{C}_\epsilon^{-1} = \sum_{k=1}^{N-1} \frac{1}{\lambda_k + \epsilon} a_k a_k^T + \sum_{k=N}^{\dim(\mathcal{H}_1)} \frac{1}{\epsilon} a_k a_k^T$ where the a_k's are the eigenvectors of $\widehat{C}$ (the first and second sums respectively gather the vectors of the range and of the nullspace of $\widehat{C}$) . As v^* lies in the range of $\widehat{C}$, it holds that $\mid v^* + \widehat{v} - \widehat{v} \mid_{\widehat{C}_\epsilon}^2 = \mid v^* \mid_{\widehat{C}_\epsilon}^2 = \sum_{k=1}^{N-1} \frac{1}{\lambda_k + \epsilon} (a_k^T v^*)^2$. Now as the a_k's do not depend on ϵ, by letting ϵ tending to zero, this quantity will tend to

$$
\sum_{k=1}^{N-1} \frac{1}{\lambda_k} (a_k^T v^*)^2 = \mid v^* \mid_{\widehat{C}}^2 = \mid v^* + \widehat{v} - \widehat{v} \mid_{\widehat{C}}^2 .
$$

Therefore it holds that $J_\epsilon(v^*+\widehat{v}) \underset{\epsilon\to 0}{\to} J(v^*+\widehat{v})$. From this we deduce that there exists $\delta > 0$ such that for all $0<\epsilon<\delta$, $J_\epsilon(v_\epsilon) \leq J(v^*+\widehat{v})+1$.

Then set $w_\epsilon = v_\epsilon - \widehat{v} = w_\epsilon^0 + w_\epsilon^1$ where w_ϵ^0 lies in the nullspace of $\widehat{C}$ and w_ϵ^1 in its range (recall that for a symmetric matrix nullspace and range are orthogonal) and see that $J_\epsilon(v_\epsilon) = \frac{1}{2} \mid y' - Hw_\epsilon \mid_\Gamma^2 + \frac{1}{2} \mid w_\epsilon \mid_{\widehat{C}_\epsilon}^2$. It holds that $\frac{1}{2} \mid w_\epsilon \mid_{\widehat{C}_\epsilon}^2 \leq J_\epsilon(v_\epsilon) \leq J(v^*+\widehat{v})+1$ for ϵ sufficiently small. Furthermore $\mid w_\epsilon \mid_{\widehat{C}_\epsilon}^2 = \mid w_\epsilon^0 \mid_{\widehat{C}_\epsilon}^2 + \mid w_\epsilon^1 \mid_{\widehat{C}_\epsilon}^2 = \frac{1}{\epsilon} \mid w_\epsilon^0 \mid^2 + \mid w_\epsilon^1 \mid_{\widehat{C}_\epsilon}^2$, and since this quantity is bounded from above we deduce that $w_\epsilon^0 \underset{\epsilon\to 0}{\to} 0$ and that w_ϵ^1 is bounded. Let $(\epsilon_m)_{m\in\mathbb{N}}$ be a sequence of positive real numbers such that $\epsilon_m \underset{m\to\infty}{\to} 0$, and from the preceding extract a converging subsequence (denoted $(\epsilon_m)_{m\in\mathbb{N}}$ for simplicity) such that $(w_{\epsilon_m}^1)_{m\in\mathbb{N}}$ converges to a limit denoted w^*. As $w_{\epsilon_m}^1$ lies in $\mathcal{R}(\widehat{C})$, we can use the eigenvalue decomposition of $\widehat{C}$ to show that $\mid w_{\epsilon_m}^1 \mid_{\widehat{C}_{\epsilon_m}}^2 \underset{m\to\infty}{\to} \mid w^* \mid_{\widehat{C}}^2$. This limiting identity, and the fact that w_ϵ^0 has limit 0, may be used to establish the first equality within the following chain of equalities and inequalities:

$$J(w^*+\widehat{v}) = \lim_{m\to\infty} \frac{1}{2} \mid y' - Hw_{\epsilon_m} \mid_\Gamma^2 + \frac{1}{2} \mid w_{\epsilon_m}^1 \mid_{\widehat{C}_{\epsilon_m}}^2 \leq \lim_{m\to\infty} J_{\epsilon_m}(v_{\epsilon_m})$$
$$\leq \lim_{m\to\infty} J_{\epsilon_m}(v^*+\widehat{v}) = J(v^*+\widehat{v}).$$

Now note that w^* matches all the constraints of (14). Indeed $w_{\epsilon_m}^1$ lies in the range of $\widehat{C}$ which is a closed space, also $v_{\epsilon_m} - \widehat{v} = w_{\epsilon_m}^0 + w_{\epsilon_m}^1 \underset{m\to\infty}{\to} w^*$. It is clear that $v_{\epsilon_m} - \widehat{v}$ matches the equality and inequality constraints of (14) for all m and hence passing to the limit we have that w^* satisfies the equalities and inequalities.

From the uniqueness of the minimizer of (14) we have that w^* is equal to v^*. In particular this means that v^* is the unique cluster point of the original sequence $(w_{\epsilon_m}^1)_{m\in\mathbb{N}}$. Since the original sequence was arbitrarily chosen, we conclude that $\lim_{\epsilon\to 0} v_\epsilon = v^* + \widehat{v}$. □

Remark 2.6. Notice that the proof remains true if we take general convex inequalities. We simply need the constrained sets to be closed and convex; however we have restricted to linear equality and inequality constraints for simplicity and because these arise most often in practice.

3. Ensemble Kalman Inversion

3.1. Inverse Problem

In this section we show how a generic inverse problem may be formulated as a partially observed dynamical system. This enables the machinery from the preceding section 2 to be used to solve inverse problems.

We are interested in the inverse problem of finding $u \in \mathcal{H}_1$ from $y \in \mathcal{H}_2$ where

$$y = G(u) + \eta,\ \eta \sim N(0, \Gamma)\,.$$

Time does not appear (explicitly) in this equation (although G may involve solution of a time-dependent differential equation, for example). In order to use the ideas from the previous section, we introduce a new variable $w = G(u)$ and rewrite the equation as

$$\begin{aligned} w =& G(u), \\ y =& w + \eta. \end{aligned}$$

The key point about writing the equation this way is that the data y is now linearly related to the variable $v = (u, w)^T$ and now we may apply the ideas of the previous section to the model by introducing the following dynamical system, taking $y_{j+1} = y$ as the given data:

$$\begin{aligned} u_{j+1} =& u_j, \\ w_{j+1} =& G(u_j), \\ y_{j+1} =& w_{j+1} + \eta_{j+1}. \end{aligned}$$

If we introduce the new variables

$$v = (u, w)^T, \quad \Psi(v) = (u, G(u))^T \tag{16a}$$

$$H = [0, I], \quad H^{\perp} = [I, 0]\,, \tag{16b}$$

and write $v_j = (u_j, w_j)^T$, we may write the dynamical system in the form

$$v_{j+1} = \Psi(v_j) \tag{17a}$$

$$y_{j+1} = H v_{j+1} + \eta_{j+1}, \tag{17b}$$

which is exactly in the same form as in the previous section. We note that

$$Hv = w, \quad H^{\perp} v = u.$$

3.2. Ensemble Kalman Inversion

The prediction step and the Kalman gain are defined as in (3), and the solution of the optimization problem is given by (4). We now simplify these formulae using the specific structure on Ψ, v, H arising in the inverse problem and given in (16); this results in block form vectors and matrices. First we note that

$$\widehat{C}_{j+1} = \begin{bmatrix} C^{uu}_{j+1} & C^{uw}_{j+1} \\ (C^{uw}_{j+1})^T & C^{ww}_{j+1} \end{bmatrix}, \quad \bar{v}_{j+1} = \begin{pmatrix} \bar{u}_{j+1} \\ \bar{w}_{j+1} \end{pmatrix}.$$

Here

$$\bar{u}_{j+1} = \frac{1}{N}\sum_{n=1}^{N} u^{(n)}_j, \ \bar{w}_{j+1} = \frac{1}{N}\sum_{n=1}^{N} G(u^{(n)}_j) := \bar{G}_j$$

and

$$\begin{aligned} C^{uw}_{j+1} &= \frac{1}{N}\sum_{n=1}^{N}(u^{(n)}_j - \bar{u}_{j+1}) \otimes (G(u^{(n)}_j) - \bar{G}_j), \\ C^{ww}_{j+1} &= \frac{1}{N}\sum_{n=1}^{N}(G(u^{(n)}_j) - \bar{G}_j) \otimes (G(u^{(n)}_j) - \bar{G}_j), \\ C^{uu}_{j+1} &= \frac{1}{N}\sum_{n=1}^{N}(u^{(n)}_j - \bar{u}_{j+1}) \otimes (u^{(n)}_j - \bar{u}_{j+1}). \end{aligned}$$

The covariance C^{ww}_{j+1} denotes the empirical covariance of the ensemble in data space, C^{uu}_{j+1} denotes the empirical covariance of the ensemble in space of the unknown u, and C^{uw}_{j+1} denotes the empirical cross-covariance from data space to the space of the unknown.

Noting that $S_{j+1} = (C^{ww}_{j+1} + \Gamma)^{-1}$ we obtain

$$K_{j+1} = \begin{pmatrix} C^{uw}_{j+1}(C^{ww}_{j+1} + \Gamma)^{-1} \\ C^{ww}_{j+1}(C^{ww}_{j+1} + \Gamma)^{-1} \end{pmatrix}. \tag{18}$$

Combining equation (18) with the update equation within (4) it follows that

$$\{v^{(n)}_j\}_{n=1}^N \to \{v^{(n)}_{j+1}\}_{n=1}^N$$

and

$$\{H^{\perp} v^{(n)}_j\}_{n=1}^N \to \{H^{\perp} v^{(n)}_{j+1}\}_{n=1}^N$$

and hence that

$$u^{(n)}_{j+1} = H^{\perp} v^{(n)}_{j+1} = u^{(n)}_j + C^{uw}_{j+1}\Big(C^{ww}_{j+1} + \Gamma\Big)^{-1}\Big(y^{(n)}_{j+1} - G(u^{(n)}_j)\Big).$$

Thus we have derived the EKI update formula:

$$u^{(n)}_{j+1} = u^{(n)}_j + C^{uw}_{j+1}\Big(C^{ww}_{j+1} + \Gamma\Big)^{-1}\Big(y^{(n)}_{j+1} - G(u^{(n)}_j)\Big). \tag{19}$$

We note also that

$$w_{j+1}^{(n)} = G(u_j^{(n)}) + C_{j+1}^{ww}\Big(C_{j+1}^{ww} + \Gamma\Big)^{-1}\Big(y_{j+1}^{(n)} - G(u_j^{(n)})\Big). \tag{20}$$

However $w_{j+1}^{(n)}$ is not needed to update the state and so plays no role in this unconstrained EKI algorithm. (It may be used, however, to impose constraints on observation space, as discussed in the next subsection.)

In summary we have derived the following algorithm for solution of the unconstrained inverse problem:

Algorithm 5 EKI Algorithm

```
1: Choose $\{u_0^{(n)}\}_{n=1}^N$, $j = 0$
2: Calculate forward model applications $\{G(u_j^{(n)})\}_{n=1}^N$
3: Update $\{u_{j+1}^{(n)}\}_{n=1}^N$ from (19)
4: $j \leftarrow j + 1$, go to 2.
```

3.3. *Ensemble Kalman Inversion With Constraints*

3.3.1. Formulation In The Original Variables We now consider imposing constraints on the optimization step arising in ensemble Kalman inversion. As in the unconstrained case we do this by formulating the problem as a special case of the partially observed dynamical system, subject to constraints, from the previous section.

To this end we formulate the constraints in the space of the unknown and the data as follows:

$$F^u u = f^u, \tag{21a}$$

$$F^w w = f^w, \tag{21b}$$

$$G^u u \preceq g^u, \tag{21c}$$

$$G^w w \preceq g^w. \tag{21d}$$

The algorithm proceeds by predicting according to equation (1), and then optimizing (3), all using the specific structure (16), and with the optimization subject to the constraints (21), written in the notation of the general Kalman updating formulae in (23), detailed below; in particular the rewrite (23) of the constraints expresses everything in terms of the variable v. We may summarize the constraints as follows, to allow direct application of the ideas of the previous section. To this end define

$$F = \begin{pmatrix} F^u H^\perp \\ F^w H \end{pmatrix} = \begin{pmatrix} F^u & 0 \\ 0 & F^w \end{pmatrix} \tag{22a}$$

$$G = \begin{pmatrix} G^u H^\perp \\ G^w H \end{pmatrix} = \begin{pmatrix} G^u & 0 \\ 0 & G^w \end{pmatrix} \tag{22b}$$

$$f = \begin{pmatrix} f^u \\ f^w \end{pmatrix}, \; g = \begin{pmatrix} g^u \\ g^w \end{pmatrix}. \tag{22c}$$

Then the constraints (21) may be written as

$$Fv = f, \tag{23a}$$

$$Gv \preceq g. \tag{23b}$$

See Algorithm 6 for the resulting pseudo-code.

Algorithm 6 Constrained EKI Algorithm

```
 1: Choose $\{u_0^{(n)}\}_{n=1}^N$, $j = 0$
 2: Calculate forward model application $\{G(u_j^{(n)})\}_{n=1}^N$
 3: Update $\{u_{j+1}^{(n)}\}_{n=1}^N$ from (19)
 4: Update $\{w_{j+1}^{(n)}\}_{n=1}^N$ from (20)
 5: for $n = 1 : N$
 6:     if $v_{j+1}^{(n)} = (u_{j+1}^{(n)}, w_{j+1}^{(n)})$ violates constraints in (23)
 7:         $v_{j+1}^{(n)} \leftarrow$ argmin of (3) subject to (16), (23)
 8:     end if
 9: end for
10: Extract $u_{j+1}^{(n)} = H^\perp v_{j+1}^{(n)}$.
11: $j \leftarrow j + 1$, go to 2.
```

3.3.2. Formulation In Range Of The Covariance We describe an alternative way to approach the derivation of the EKI update formulae. We apply Theorem 2.4 with the specific structure (16), (23) arising from the dynamical system used in

EKI. To this end we define

$$J_{\text{filter},j,n}(b) := \frac{1}{2} \mid y_{j+1}^{(n)} - G(u_j^{(n)}) - B^w b \mid_\Gamma^2 + \frac{1}{2N}|b|^2 \tag{24a}$$

$$= \frac{1}{2} b^T \Big((B^w)^T \Gamma^{-1} B^w + \frac{1}{N} I \Big) b - \Big((B^w)^T \Gamma^{-1} (y_{j+1}^{(n)} - G(u_j^{(n)})) \Big)^T b + \text{const.} \tag{24b}$$

where b is the vector of N scalar weights b_m and

$$B^u b = \frac{1}{N} \sum_{m=1}^{N} b_m \Big(u_j^{(m)} - \bar{u}_j \Big), \tag{25a}$$

$$B^w b = \frac{1}{N} \sum_{m=1}^{N} b_m \Big(G(u_j^{(m)}) - \bar{G}_j \Big), \tag{25b}$$

$$Bb = \begin{pmatrix} B^u b \\ B^w b \end{pmatrix}. \tag{25c}$$

Once this quadratic form has been minimized with respect to b then the update formula (7) gives

$$u_{j+1}^{(n)} = u_j^{(n)} + \frac{1}{N} \sum_{m=1}^{N} b_m \Big(u_j^{(m)} - \bar{u}_j \Big), \tag{26a}$$

$$w_{j+1}^{(n)} = G(u_j^{(n)}) + \frac{1}{N} \sum_{m=1}^{N} b_m \Big(G(u_j^{(m)}) - \bar{G}_j \Big). \tag{26b}$$

Note that the vector $\{b_m\}$ depends on the particle label n; as in the previous section, we have suppressed this dependence for notational convenience. We may now impose linear equality and inequality constraints on both u and $w = G(u)$ (i.e. in parameter and data spaces) and minimize (24) subject to these constraints. To be more specific if we impose the constraints (23) expressed in the variable b:

$$FBb = f - F\widehat{v}_{j+1}^{(n)}, \tag{27a}$$

$$GBb \preceq g - G\widehat{v}_{j+1}^{(n)}. \tag{27b}$$

Here F, G, f and g are given by (22), B is defined by (25) and

$$\widehat{v}_{j+1}^{(n)} = \begin{pmatrix} u_j^{(n)} \\ G(u_j^{(n)}) \end{pmatrix}.$$

See Algorithm 7 for the resulting pseudo-code.

Algorithm 7 Constrained EKI algorithm formulated in range of covariance

```
1: Choose $\{u_0^{(n)}\}_{n=1}^N$, $j = 0$
2: Calculate forward model application $\{G(u_j^{(n)})\}_{n=1}^N$
3: Update $b^{(n)} \leftarrow$ argmin of (24), $\{u_{j+1}^{(n)}\}_{n=1}^N$ and $\{w_{j+1}^{(n)}\}_{n=1}^N$ from (26)
4: for $n = 1 : N$
5:     if $v_{j+1}^{(n)} = (u_{j+1}^{(n)}, w_{j+1}^{(n)})$ violates constraints in (27)
6:         $b^{(n)} \leftarrow$ argmin of (24) subject to (27)
7:         Update $\{u_{j+1}^{(n)}\}$ and $\{w_{j+1}^{(n)}\}$ from (26)
8:     end if
9: end for
10: $j \leftarrow j + 1$, go to 2.
```

Remark 3.1. As in the previous section, the result holds true for general convex inequality constraints; the linear case is considered for simplicity of exposition, and because it is most frequently arising in practice.

Remark 3.2. The EKI algorithm, with or without constraints, has the following invariant subspace property: define $\mathcal{A} = span(u_0^{(n)})_{n\in\{1,\cdots,N\}}$, then for all j in $\{0,\ldots,J\}$ and for all n in $\{1,\cdots,N\}$, then the $u_j^{(n)}$ defined by the three algorithms in this section all lie in $\mathcal{A}$. This is a direct consequence of writing the update formulae in terms of b and noting (26).

We can now state a result analogous to Theorem 2.4, and with proof that is a straightforward corollary of that result, using the specific structure (16):

Theorem 3.3. *Suppose that the dimensions of $\mathcal{H}_1$ and $\mathcal{H}_2$ are finite. Suppose also that the specific structure* (16) *is applied. The problem of finding $u_{j+1}^{(n)}$ from the minimizer of $I_{\text{filter},j,n}(v)$, defined in* (3) *and subject to the constraint* (21)*, is equivalent to finding b that minimizes* (24)*, subject to* (27)*, and then using* (26) *to find $u_{j+1}^{(n)}$ from b. Furthermore, both of these constrained minimization problems have a unique solution provided that the constraint sets are non-empty.*

4. Numerical Results

This section contains numerical results which demonstrate the benefits of imposing constraints on ensemble Kalman methods. Subsection 4.1 concerns an application of state estimation (using EnKF) in biomedicine, using real patient data, whilst subsection 4.2 concerns on application of inversion (using EKI) in seismology and employs simulated data. When comparing results from the two experiments, recall that iterations of EKI correspond to an algorithmic dynamics intended to converge to a single distribution (over ensemble members) on the parameters for which we invert, whereas iterations of EnKF correspond to the incorporation of new data at every physical measurement time, and thus the distribution (over ensemble members) is not necessarily expected to converge as the iteration progresses.

4.1. State Estimation

Here we present an application of the constrained EnKF to the tracking and forecasting of human blood glucose levels. We use self-monitoring data collected by an individual with Type 2 Diabetes. We use the "P1" data set described by Albers *et al.* in [32]; this dataset includes measurements of blood glucose and consumed nutrition, and is publicly available on `physionet.org`. For more information on the data, and on an unconstrained data assimilation approach using the unscented Kalman filter, see [32]. We model the glucose-insulin system with the ultradian model proposed by [33]. The primary state variables are the glucose concentration, G, the plasma insulin concentration, I_p, and the interstitial insulin concentration, I_i; these three state variables are augmented with a three stage delay (h_1, h_2, h_3) which encodes a non-linear delayed hepatic glucose response to plasma insulin levels. The resulting ordinary differential equations have the form:

$$\frac{dI_p}{dt} = f_1(G) - E(\frac{I_p}{V_p} - \frac{I_i}{V_i}) - \frac{I_p}{t_p} \tag{28a}$$

$$\frac{dI_i}{dt} = E(\frac{I_p}{V_p} - \frac{I_i}{V_i}) - \frac{I_i}{t_i} \tag{28b}$$

$$\frac{dG}{dt} = f_4(h_3) + m_G(t) - f_2(G) - f_3(I_i)G \tag{28c}$$

$$\frac{dh_1}{dt} = \frac{1}{t_d}(I_p - h_1) \tag{28d}$$

$$\frac{dh_2}{dt} = \frac{1}{t_d}(h_1 - h_2) \tag{28e}$$

$$\frac{dh_3}{dt} = \frac{1}{t_d}(h_2 - h_3) \tag{28f}$$

Here $m_G(t)$ represents a known rate of ingested carbohydrates, $f_1(G)$ represents the rate of glucose-dependent insulin production, $f_2(G)$ represents insulin-independent glucose utilization, $f_3(I_i)G$ represents insulin-dependent glucose utilization and $f_4(h_3)$ represents delayed insulin-dependent hepatic glucose production; the functional forms of these parameterized processes can be found in the appendix, along with a description of model parameters.

In the EnKF setting, we write $u = [I_p, I_i, G, h_1, h_2, h_3]$, and use (28) to define F such that

$$\frac{du}{dt} = F(u, t, \theta),$$

where θ contains model parameters. We then extend the state vector in order to perform joint parameter estimation: $v = [u, R_g]^T$.

For the purposes of this paper, the function $m_G(t)$ may be viewed as known; it is determined from data describing meals consumed by the patient. Since insulin (I_p and I_i) and delay variables (h_1, h_2, and h_3) are not measured, whilst glucose is measured, we define the measurement operator to be $H = [0, 0, 1, 0, 0, 0, 0]$. The discrete time forward model is obtained by integrating the deterministic model in (28) between consecutive measurement time-points and applying an identity map to R_g. Because these time-points may not be equally spaced, and because the time-dependent forcings (meals) will differ in different time-intervals, this leads to a map of the form

$$v_{j+1} = \Psi_j(v_j).$$

This is a slight departure from the methodology outlined in section 2, where Ψ does not depend on j (autonomous dynamics) but is a straightforward extension which the reader can easily provide.

We present EnKF results from a single patient's data when run with and without constraints (Algorithms 1 and 3 respectively). We performed joint state-parameter estimation, augmenting the state with parameter R_g (see Appendix for details of where this parameter appears) and adding identity-map dynamics for parameter R_g. The following constraints were imposed:

$$\begin{bmatrix} 0.01 \\ 0.01 \\ 2000 \\ 0.01 \\ 0.01 \\ 0.01 \\ 0 \end{bmatrix} \preceq v \preceq \begin{bmatrix} 10000 \\ 10000 \\ 40000 \\ 10000 \\ 10000 \\ 10000 \\ 1000000 \end{bmatrix} \tag{29}$$

Figure 1 compares the overall distribution of updated state means over time when running EnKF with and without these state constraints. While individual particles in this experiment often violated the constraints, the overall updated means did not. Nevertheless, enforcement of lower-bound constraints shifts up the state distribution slightly. Note that upper bound constraints were never violated in this experiment.

Figure 2 shows a two-dimensional state projection of updated particles at a given time step before and after applying the constrained optimization. Note that particles may additionally violate constraints in unplotted dimensions—this explains why one particle whose unconstrained update appears to live within the constraints is in fact differently updated under the constrained optimization. Time step 126 was selected for illustrative purposes, and was the measurement event in which particles most often violated the constraints.

Figure 3 depicts the overall frequency of constraint violations. We observe that the the measured state (blood glucose) never violated a constraint, nor did the inferred parameter R_g. However, other model states did often violate constraints, and up to 30% (4/13) of particles simultaneously violated the constraints at a single time-step.

By adding constraints, we ensure that all the simulations which constitute the ensemble method are biologically plausible.

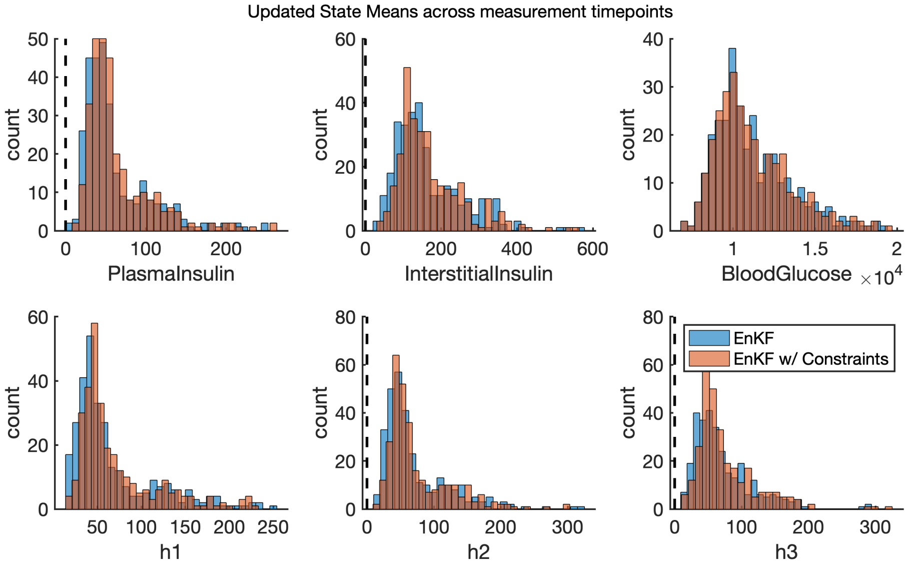


Figure 1 The distribution of mean state updates when running EnKF with and without inequality constraints. Black vertical lines denote lower bound state constraints.

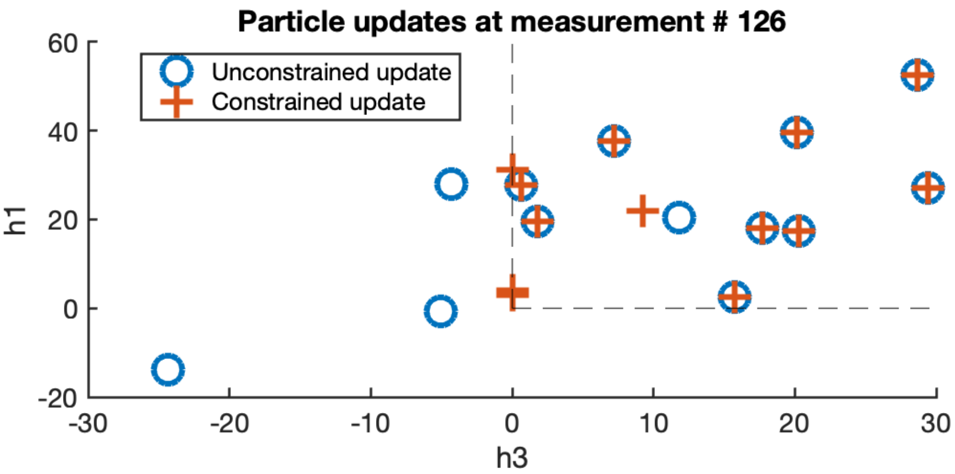


Figure 2 Particle updates at a given time-step (here, measurement 126) are shown using a traditional Kalman gain versus using the constrained optimization. The black lines denote lower bound constraints on the states h_1 and h_3.

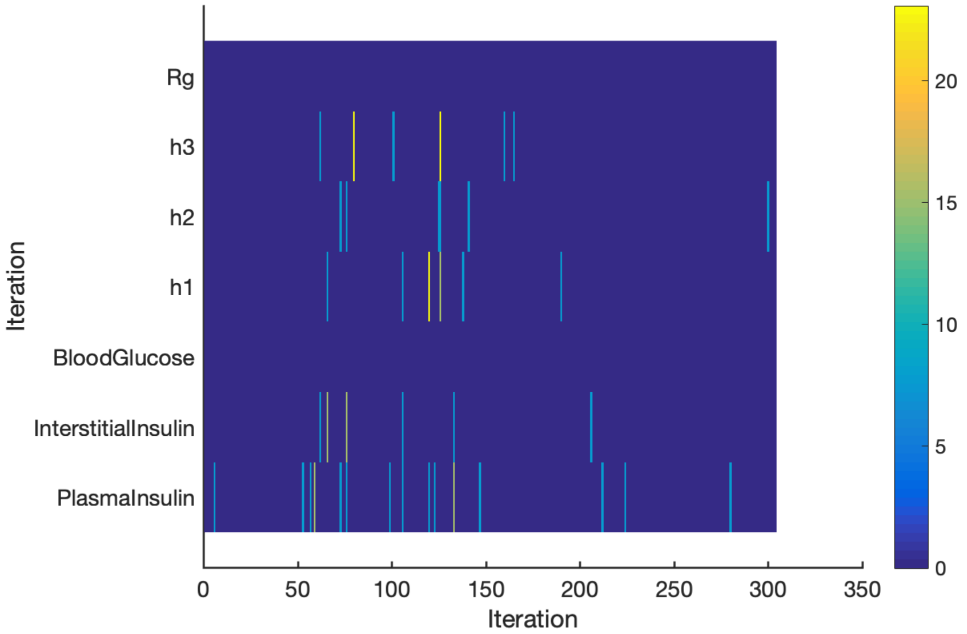


Figure 3 Percentage map of the constraint violations, where each lower-bound constraint is represented by a row. At each iteration, the percentage of particles that violated a constraint is color-coded, with yellow representing the largest proportion of constraint violations.

4.2. Inverse Problem

Here we present application of the constrained EKI in seismology. We study near-surface site characterization in which we invert for the shear wave velocity profile of the geomaterials in the earth shallow crust, using downhole array data. For forward modeling, we consider a semi-discrete form of the following wave equation in a horizontally stratified heterogeneous soil layer:

$$\frac{\partial}{\partial z}\left[c_s^2(z)\frac{\partial d(z,t)}{\partial z}\right]-\frac{\partial^2 d(z,t)}{\partial t^2}=0.$$

Here $d(z,t)$ is the displacement field of the wave response as a function of spatial variable $z \in (0,H)$ and time variable $t \in (0,T]$. The function $c_s(z)$ is the shear wave velocity function. We impose the following boundary and initial conditions:

$$d(H,t)=d_0(t), \quad \partial d(0,t)/\partial z=0, \quad d(z,0)=0, \quad \partial d(z,0)/\partial t=0$$

where $d_0(t)$ is the prescribed displacement at depth $z = H$. Generally, the shear wave velocity changes as a piecewise constant function with depth. If the layering information, i.e., the total number of layers and their thickness, is not available or is poorly characterized, it is desired to use a generic function for site characterization, such as this:

$$c_s(z)=\begin{cases} c_{s0} & 0 \le z \le z_0 \\ c_{s0}(1+k(z-z_0))^n & z_0 \le z \le z_1 \\ \alpha c_{s0}(1+k(z_1-z_0))^n & z_1 \le z \le H \end{cases}.$$

See, for example, [34]. In the constrained EKI setting, $u = (c_{s0}, k, z_0, n, z_1, \alpha)$ and

$$G(u) = \partial^2 d(0,t)/\partial t^2.$$

For the numerical example studied here, G^u and g^u are determined by enforcing the constraints $0 \le c_{s0} \le 1000$, $0 \le k \le 100$, $0 \le z_0 \le z_1$, $0 \le n \le 1$, $z_0 \le z_1 \le H$, and $1 \le \alpha \le 10$. We generate the initial ensemble by drawing samples from uniform distributions and discard members that violate the enforced constraints. In order to avoid very large velocities at $z = z_1$, we also discard members with $c_s(z_1) > 5000$m/s. If we perform parameter learning using the unconstrained EKI, the experiment fails at $j = 1$ because of incapability of the dynamic model to propagate unphysical values of the shear wave velocity c_s.

All results shown use Algorithm 7. Figure 4 shows the ensemble distribution of u at $j = 2$ before and after enforcing constraints whilst Figure 5 shows the evolution of the updated ensemble. Note that parameter k saturates with an ensemble close to the upper bound of 100 imposed through constraints on this parameter; however experiments in which we imposed different upper bounds on this parameter lead to different estimates for k, with little change to the estimated velocity profile and we conclude that this parameter suffers from identifiability issues. (Note that Figure 4 displays the updated ensemble distribution at a single step in the sequence of ensemble updates, comparing the effect of imposing constraints with neglecting them; in contrast Figure 1 shows the distribution over all measurement time-points of the ensemble means. The figures thus illustrate different phenomena).

Moreover, Figure 6a shows the map of violation for different constraints enforced on parameters whilst Figure 6b shows the estimated generic c_s profile after 40 iterations compared to the true profile and the initial estimate. Figure 6a shows the key role employed by the enforcing of constraints. In this case the addition of constraints ensures that all the simulations which constitute the ensemble method are physically meaningful, and also that the forward model remains well-posed.

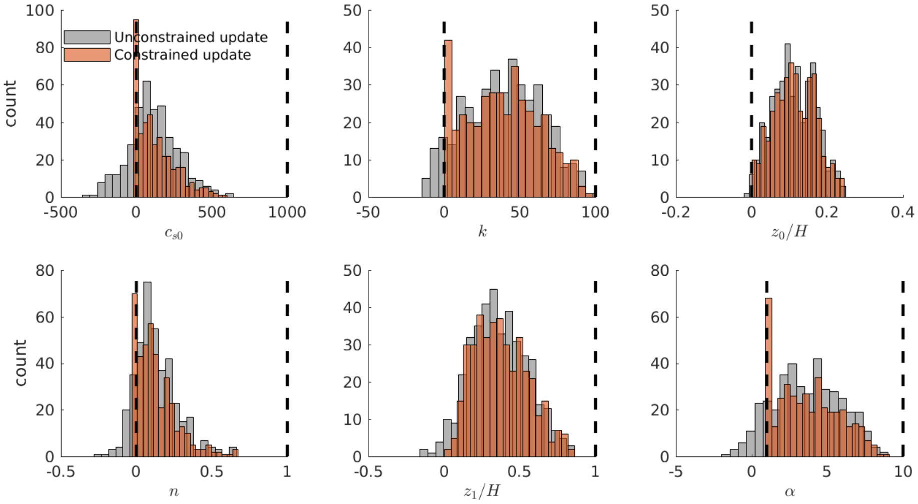


Figure 4 The distribution of parameters before and after enforcing constraints in Algorithm 7 at iteration $j = 2$. Black vertical lines denote the lower and upper bound constraints.

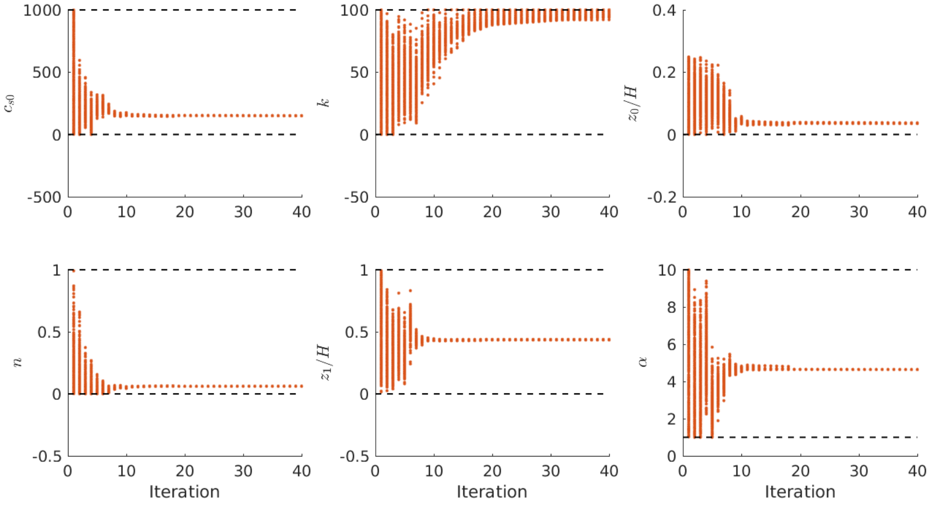
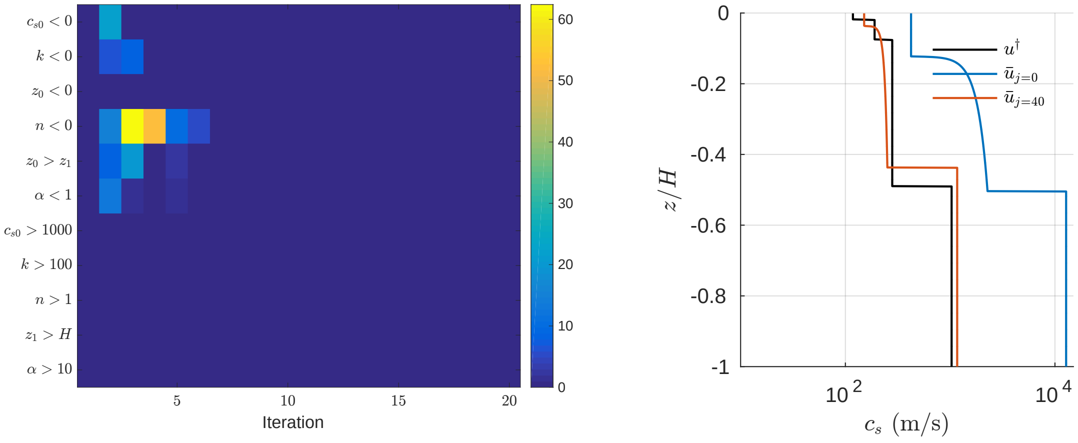


Figure 5 Evolution of the updated ensemble with iteration. Black horizontal lines denote the lower and upper bound constraints.



(a)

(b)

Figure 6 (a) The percentage map of the constraint violations for the first 20 iterations; (b) the estimated velocity profile ($\bar{u}_{j=40}$) compared to the true profile ($u^\dagger$) and the initial estimate ($\bar{u}_{j=0}$)

5. Conclusions

Constraints arise naturally in many state and parameter estimation problems. We have shown how convex constraints may be incorporated into ensemble Kalman based state or parameter estimation algorithms with relatively few changes to existing code: the standard algorithm is applied and for any ensemble member which violates a constraint, a quadratic optimization problem subject to convex constraints is solved instead. We have written the resulting algorithms in easily digested pseudo-code, we have developed an underpinning theory and we have given illustrative numerical examples.

Two primary directions suggest themselves in this area. The first is the use of these methods in applications. As indicated in the introduction, our general formulation is inspired by the two papers [8, 6] from the geosciences and we have demonstrated applicability to problems from biomedicine and seismology; but many other potential application domains are ripe for application of ensemble Kalman methodology, because of its black-box and derivative-free formulation, and the ability to impose constraints in a straightforward fashion will help to extend this methodology. The second is the theoretical analysis of these methods: can the inclusion of constraints be used to deduce improved accuracy of state or parameter estimates; or can the inclusion of constraints be used to demonstrate improved performance as measured, for example, by proportion of model runs which are physically (or biologically etc.) plausible? Furthermore, although the imposition of constraints is reasonable, it is not clear that it may not lead to pathologies in algorithmic performance and ruling out, or understanding, the occurrence of such pathological behaviour may be important.

Acknowledgments

This work was funded by NIH-NLM grant RO1 LM012734. AMS was also funded by AFOSR Grant FA9550-17-1-0185 and by ONR grant N00014-17-1-2079.

Appendix

We give the details of the ultradian model of glucose-insulin dynamics used as the forward model in subsection 4.1. An example of the induced dynamics is given in Figure 7.

$$\frac{dI_p}{dt} = f_1(G) - E(\frac{I_p}{V_p} - \frac{I_i}{V_i}) - \frac{I_p}{t_p} \tag{30}$$

$$\frac{dI_i}{dt} = E(\frac{I_p}{V_p} - \frac{I_i}{V_i}) - \frac{I_i}{t_i} \tag{31}$$

$$\frac{dG}{dt} = f_4(h_3) + m_G(t) - f_2(G) - f_3(I_i)G \tag{32}$$

$$\frac{dh_1}{dt} = \frac{1}{t_d}(I_p - h_1) \tag{33}$$

$$\frac{dh_2}{dt} = \frac{1}{t_d}(h_1 - h_2) \tag{34}$$

$$\frac{dh_3}{dt} = \frac{1}{t_d}(h_2 - h_3) \tag{35}$$

where, for N meals at times $\{t_j\}_{j=1}^N$ with carbohydrate composition $\{m_j\}_{j=1}^N$

$$m_G(t) = \sum_{j=1}^{N} \frac{m_j k}{60} \exp(k(t_j - t)), \quad N = \#\{t_j < t\} \tag{36}$$

and

$$f_1(G) = \frac{R_m}{1 + \exp(\frac{-G}{V_g c_1} + a_1)} \text{ : the rate of insulin production} \tag{37}$$

$$f_2(G) = U_b(1 - \exp(\frac{-G}{C_2 V_g})) \text{ : insulin-independent glucose utilization} \tag{38}$$

$$f_3(I_i) = \frac{1}{C_3 V_g}(U_0 + \frac{U_m - U_0}{1 + (\kappa I_i)^{-\beta}}), \;\; f_3(I_i)G \text{ : insulin-dependent glucose utilization} \tag{39}$$

$$f_4(h_3) = \frac{R_g}{1 + \exp(\alpha(\frac{h_3}{C_5 V_p} - 1))} \text{ : delayed insulin-dependent glucose utilization} \tag{40}$$

$$\kappa = \frac{1}{C_4}(\frac{1}{V_i} - \frac{1}{E t_i}) \tag{41}$$

References

[1] Teixeira B O, Tôrres L A, Aguirre L A and Bernstein D S 2010 *Journal of Process Control* **20** 45–57

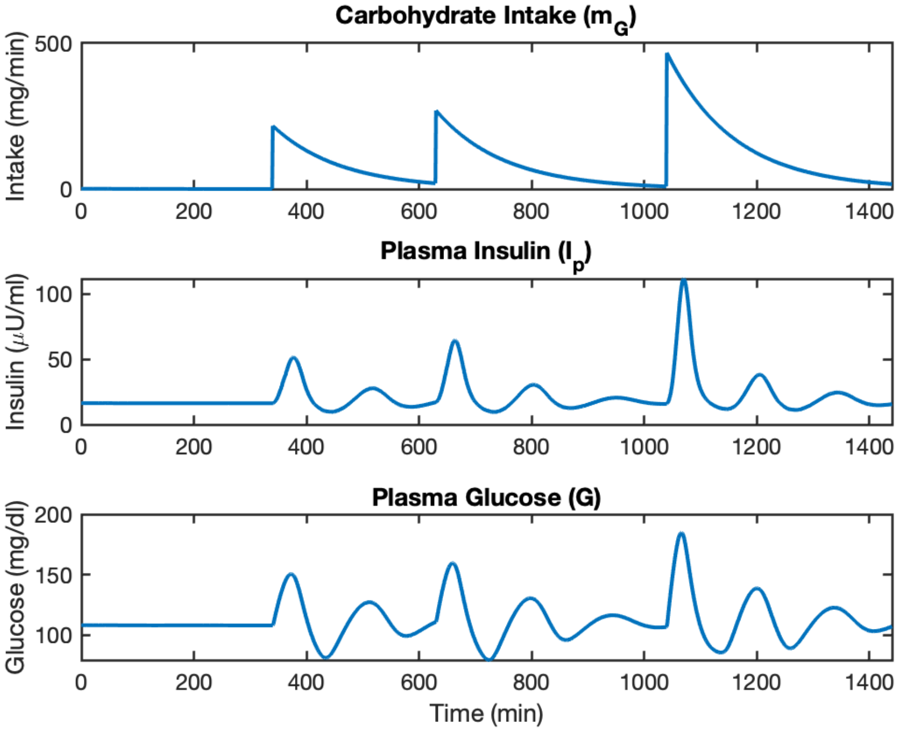


Figure 7 Here we show the oscillating dynamics of the glucose-insulin response in the ultradian model, driven by an exponentially decaying nutritional driver m_G.

[2] Bonnet V, Dumas R, Cappozzo A, Joukov V, Daune G, Kulić D, Fraisse P, Andary S and Venture G 2017 *Journal of biomechanics* **62** 140–147

[3] Goffaux G, Perrier M and Cloutier M 2011 Cell energy metabolism: a constrained ensemble kalman filter *Proceedings of the 18th IFAC world congress: Milano, Italy, International Federation of Automatic Control* pp 8391–8396

[4] Lei J, Liu S and Wang X 2012 *IET Science, Measurement & Technology* **6** 63–77

[5] Simon D and Simon D L 2010 *International Journal of Systems Science* **41** 159–171

[6] Janjić T, McLaughlin D, Cohn S E and Verlaan M 2014 *Monthly Weather Review* **142** 755–773

[7] Yang X, Huang B and Prasad V 2014 *Chemical Engineering Science* **106** 211–221

[8] Wang D, Chen Y and Cai X 2009 *Water resources research* **45**

[9] Evensen G 2009 *Data assimilation: the ensemble Kalman filter* (Springer Science & Business Media)

[10] Reich S and Cotter C 2015 *Probabilistic forecasting and Bayesian data assimilation* (Cambridge University Press)

[11] Law K, Stuart A and Zygalakis K 2015 *Data Assimilation* (Springer)

[12] Carrassi A, Bocquet M, Bertino L and Evensen G *Data assimilation in the geosciences: An overview of methods, issues, and perspectives* 5 (Wiley Interdisciplinary Reviews: Climate

Change, 5(2018))

[13] Evensen G 1994 *Journal of Geophysical Research: Oceans* **99** 10143–10162

[14] Burgers G, Jan van Leeuwen P and Evensen G 1998 *Monthly weather review* **126** 1719–1724

[15] Oliver D S, Reynolds A C and Liu N 2008 *Inverse theory for petroleum reservoir characterization and history matching* (Cambridge University Press)

[16] Iglesias M A, Law K J and Stuart A M 2013 *Inverse Problems* **29** 045001

[17] Doucet A, De Freitas N and Gordon N 2001 An introduction to sequential monte carlo methods *Sequential Monte Carlo methods in practice* (Springer) pp 3–14

[18] Del Moral P, Doucet A and Jasra A 2006 *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **68** 411–436

[19] Simon D 2010 *IET Control Theory & Applications* **4** 1303–1318

[20] Rasool G 2018 Constrained state estimation — a review (*Preprint* arXiv:1807.03463)

[21] Robertson D G, Lee J H and Rawlings J B 1996 *AIChE Journal* **42** 2209–2224

[22] Rao C V, Rawlings J B and Mayne D Q 2003 *IEEE transactions on automatic control* **48** 246–258

[23] Vachhani P, Rengaswamy R, Gangwal V and Narasimhan S 2005 *AIChE Journal* **51** 946–959

[24] Li R, Jan N M, Prasad V and Huang B 2018 Constrained extended kalman filter based on kullback-leibler (kl) divergence *2018 European Control Conference (ECC)* (IEEE) pp 831–836

[25] Vachhani P, Narasimhan S and Rengaswamy R 2006 *Journal of process control* **16** 1075–1086

[26] Julier S, Uhlmann J and Durrant-Whyte H F 2000 *IEEE Transactions on automatic control* **45** 477–482

[27] Simon D and Simon D L 2006 *IEE Proceedings-Control Theory and Applications* **153** 371–378

[28] Mandela R, Kuppuraj V, Rengaswamy R and Narasimhan S 2012 *Journal of Process Control* **22** 718–728

[29] Prakash J, Patwardhan S C and Shah S L 2008 *2008 American Control Conference* 3542–3547

[30] Prakash J, Patwardhan S C and Shah S L 2010 *Industrial & Engineering Chemistry Research* **49** 2242–2253

[31] Stuart A and Zygalakis K 2015 Data assimilation: A mathematical introduction Tech. rep. Oak Ridge National Laboratory (ORNL), Oak Ridge, TN (United States)

[32] Albers D J, Levine M, Gluckman B, Ginsberg H, Hripcsak G and Mamykina L 2017 *PLoS computational biology* **13** e1005232

[33] Sturis J, Polonsky K S, Mosekilde E and Van Cauter E 1991 *American Journal of Physiology-Endocrinology And Metabolism* **260** E801–E809

[34] Shi J and Asimaki D 2018 *Seismological Research Letters* **89** 1397–1409