Ensemble Kalman Methods for High-Dimensional Hierarchical Dynamic Space-Time Models

Matthias Katzfuss, Jonathan R. Stroud & Christopher K. Wikle

Taylor & Francis
Taylor & Francis Group

Check for updates

Ensemble Kalman Methods for High-Dimensional Hierarchical Dynamic Space-Time Models

Matthias Katzfuss[a], Jonathan R. Stroud[b], and Christopher K. Wikle[c]

[a]Department of Statistics, Texas A&M University, College Station, TX; [b]McDonough School of Business, Georgetown University, Washington, DC; [c]Department of Statistics, University of Missouri, Columbia, MO

ABSTRACT

We propose a new class of filtering and smoothing methods for inference in high-dimensional, nonlinear, non-Gaussian, spatio-temporal state-space models. The main idea is to combine the ensemble Kalman filter and smoother, developed in the geophysics literature, with state-space algorithms from the statistics literature. Our algorithms address a variety of estimation scenarios, including online and off-line state and parameter estimation. We take a Bayesian perspective, for which the goal is to generate samples from the joint posterior distribution of states and parameters. The key benefit of our approach is the use of ensemble Kalman methods for dimension reduction, which allows inference for high-dimensional state vectors. We compare our methods to existing ones, including ensemble Kalman filters, particle filters, and particle MCMC. Using a real data example of cloud motion and data simulated under a number of nonlinear and non-Gaussian scenarios, we show that our approaches outperform these existing methods. Supplementary materials for this article are available online.

1. Introduction

1.1. Inference in State-Space Models

State-space models (SSMs) are general representations of systems observed over time. A SSM describes the temporal evolution of the system state, and the relationship of the state to observations (e.g., West and Harrison 1997; Shumway and Stoffer 2006). Here, we analyze spatio-temporal SSMs describing the evolution of one or more discretized spatial fields over time. These models arise, for example, in geoscience and biomedical applications. They are often nonlinear, non-Gaussian, and have very high-dimensional states. Standard statistical approaches such as Markov chain Monte Carlo (MCMC) and particle filters break down for this class of models. Thus, new methods are needed for accurate inference. We propose a number of new filtering and smoothing algorithms for Bayesian inference in these high-dimensional SSMs.

There is an enormous statistics and engineering literature on state and parameter estimation for low-dimensional SSMs. For linear, Gaussian models, the Kalman filter (Kalman 1960) provides closed-form posterior distributions for the states conditional on the parameters. For conditionally Gaussian models, inference is often carried out using *MCMC methods* based on the forward-filter backward-sampling algorithm (Carter and Kohn 1994; Frühwirth-Schnatter 1994) or the mixture Kalman filter (Chen and Liu 2000). MCMC methods have also been developed for more general SSMs, including nonlinear, non-Gaussian models (Carlin, Polson, and Stoffer 1992) and non-Gaussian measurement models (Shephard and

Pitt 1997; Gamerman 1998). However, these methods do not scale to high-dimensional states because their computational cost grows cubically in the state dimension.

Another popular approach for nonlinear, non-Gaussian SSMs is sequential Monte Carlo (SMC) methods, also known as particle filters (e.g., Doucet, De Freitas, and Gordon 2001). SMC methods approximate the state distribution at each time by a weighted set of samples or *particles*. These particles are propagated forward through time according to the state evolution. They are then updated based on new data using a reweighting or resampling step (e.g., Gordon, Salmond, and Smith 1993; Liu and Chen 1998). Although these methods are exact with an infinite number of particles, it is well known that they suffer from weight degeneracy (i.e., all weights but one become essentially zero) in high-dimensional problems (e.g., Snyder et al. 2008). Thus, particle filters do not scale to high dimensions.

In contrast to the statistical approaches above, approximate state-estimation methods in the geoscience literature scale well to high-dimensional problems. In this paper, we focus on a geophysical data assimilation method called the *ensemble Kalman filter* (EnKF; Evensen 1994); see Houtekamer and Zhang (2016) and Katzfuss, Stroud, and Wikle (2016) for recent reviews, and Wikle and Berliner (2007) for a general introduction to data assimilation. The EnKF also uses an ensemble representation that is propagated forward like the particle filter, but instead of reweighting, the EnKF updates the ensemble using a linear "shift" that approximates the best linear update. While the EnKF is based on the assumption of a linear, Gaussian model, it

provides accurate inference for many nonlinear models as well. This has allowed the EnKF to be successfully applied to non-linear spatio-temporal SSMs in millions of dimensions (e.g., Szunyogh et al. 2008; Whitaker et al. 2008; Houtekamer et al. 2014).

The EnKF is also used for parameter estimation and non-Gaussian data assimilation. The most common EnKF method for parameter estimation is *state augmentation* (Anderson 2001), which refers to simply including the parameters in the state vector. While EnKF with state augmentation often works well for certain parameters, such as autoregressive coefficients, the resulting linear update can be highly problematic for several types of parameters, most notably, variance and covariance parameters (Stroud and Bengtsson 2007; DelSole and Yang 2010). The linear shifting update in the EnKF can also be highly suboptimal for non-Gaussian observations. Alternative non-Gaussian EnKF approaches, including quantile-based filters (Anderson 2010) and moment-matching filters (Lei and Bickel 2011), still perform poorly for certain classes of measurement distributions. Thus, new EnKF methods are needed for more general SSMs with unknown parameters and non-Gaussian measurements.

1.2. Hierarchical SSMs

Here, we develop new methods to analyze a general class of high-dimensional *hierarchical SSMs* (HSSMs; e.g., Gamerman and Migon 1993; Cressie and Wikle 2011). These models consist of four levels, adding two levels to a standard Gaussian SSM: a *transformation level*, which allows for non-Gaussian observations, and a *parameter level*, which models the unknown parameters, assumed to be random. This class of HSSMs is very general and can be used to describe most dynamic spatio-temporal systems observed sequentially over time. It allows for unknown time-varying parameters in any part of the model, non-Gaussian and nonlinear observations, mixture models, and even higher order Markov models by extending the state-space.

More specifically, for discrete time points $t = 1, 2, \ldots, T$, our HSSM is given by

$$\text{Transformation:} \quad \mathbf{z}_t | \mathbf{y}_t, \boldsymbol{\theta}_t \sim f_t(\mathbf{z}_t | \mathbf{y}_t, \boldsymbol{\theta}_t), \tag{1}$$

$$\text{Observation:} \quad \mathbf{y}_t | \mathbf{x}_t, \boldsymbol{\theta}_t \sim \mathcal{N}_{m_t}(\mathbf{H}_t \mathbf{x}_t, \mathbf{R}_t), \tag{2}$$

$$\text{Evolution:} \quad \mathbf{x}_t | \mathbf{x}_{t-1}, \boldsymbol{\theta}_t \sim \mathcal{N}_n(\widetilde{\mathbf{x}}_t = \mathcal{M}_t(\mathbf{x}_{t-1}), \mathbf{Q}_t), \tag{3}$$

$$\text{Parameter:} \quad \boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1} \sim p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}), \tag{4}$$

where $\mathbf{z}_t$ is the m_t-dimensional measurement vector at time t, $\mathbf{y}_t$ is a corresponding m_t-dimensional "latent observation" vector, $\mathbf{x}_t$ is the n-dimensional unobserved state vector, $\boldsymbol{\theta}_t$ is the parameter vector, f_t is a known measurement or transformation distribution, $\mathcal{M}_t$ is the state evolution operator, and $\mathcal{N}_n(\cdot, \cdot)$ denotes the n-variate normal distribution. We assume that $\mathbf{z}_t$ and $\mathbf{y}_t$ are serially independent given the states and parameters. The model is completed with initial priors $\mathbf{x}_0 \sim \mathcal{N}_n(\boldsymbol{\mu}_{0|0}, \boldsymbol{\Sigma}_{0|0})$ and $\boldsymbol{\theta}_0 \sim p(\boldsymbol{\theta}_0)$.

The state vector $\mathbf{x}_t$ consists of one or more discretized spatial fields, along with any parameters that can be estimated using state augmentation. Conversely, the parameter vector $\boldsymbol{\theta}_t$ contains any unknown quantities that *cannot* be handled well by state augmentation. The quantities $\mathbf{H}_t$, $\mathcal{M}_t$, $\mathbf{R}_t$, $\mathbf{Q}_t$, and f_t may

all depend on the parameter vector $\boldsymbol{\theta}_t$. Sometimes we make the dependence explicit by writing $\mathbf{H}_t(\boldsymbol{\theta}_t)$ and so forth. Further, we view the evolution operator $\mathcal{M}_t$ as a "black box" that might be nonlinear and expensive to evaluate or unavailable in closed form. We focus on models where n or m_t are very large, usually $n > m_t$. Sometimes, we observe $\mathbf{z}_t = \mathbf{y}_t$ so that $f(\mathbf{z}_t | \mathbf{y}_t, \boldsymbol{\theta}_t) = \delta_{\mathbf{y}_t}(\mathbf{z}_t)$, where $\delta(\cdot)$ is the Dirac delta function.

The HSSM (1)–(4) is very flexible and includes many important SSMs as special cases. If $\mathbf{y}_t$ and $\boldsymbol{\theta}_t$ are known, the model reduces to a standard Gaussian SSM given by Equations (2) and (3) (e.g., Shumway and Stoffer 2006). If the evolution is linear with $\mathcal{M}_t(\mathbf{x}_{t-1}) = \mathbf{M}_t \mathbf{x}_{t-1}$, this becomes a *conditionally Gaussian SSM* (e.g., Carter and Kohn 1994; Chen and Liu 2000), where the full conditional distributions for the states are Gaussian. The HSSM also allows for non-Gaussian and nonlinear observations through f_t in (1). This could include exponential family models (e.g., Shephard and Pitt 1997; Gamerman 1998), nonlinear mean functions (e.g., Carlin, Polson, and Stoffer 1992), or threshold models for discrete data or discrete-continuous mixtures (e.g., Sansó and Guenni 1999). Finally, the HSSM allows for unknown static or time-varying parameters at any level of the model.

Note that it may be possible to reduce our HSSM to a standard nonlinear, non-Gaussian SSM by integrating out $\mathbf{y}_t$ and defining $(\mathbf{x}_t, \boldsymbol{\theta}_t)$ as the state vector. However, we prefer the hierarchical formulation in Equations (1)–(4), because it facilitates model building and computation. Specifically, by conditioning on $\mathbf{y}_t$ and $\boldsymbol{\theta}_t$, we will use the EnKF for inference on the high-dimensional states.

1.2.1. Examples
We now present some examples of the HSSM in Equations (1)–(4) that will be used for numerical comparisons in Section 5.

Example 1 (Heavy-tailed-data model). The distribution of $\mathbf{y}_t | \mathbf{x}_t$ belongs to the class of normal-scale mixtures (Andrews and Mallows 1974; West 1987) if we assume that $\mathbf{R}_t$ is diagonal with independent random parameters $\boldsymbol{\theta}_t$ on the diagonal. An important member of this class is the t-distribution, which is similar to a normal distribution but has heavier tails and is thus more robust to outliers. The following HSSM assumes that the observation noise is independently t-distributed with κ degrees of freedom:

$$\mathbf{z}_t | \mathbf{y}_t \sim \delta_{\mathbf{y}_t}(\mathbf{z}_t),$$
$$\mathbf{y}_t | \mathbf{x}_t, \boldsymbol{\theta}_t \sim \mathcal{N}_{m_t}(\mathbf{H}_t \mathbf{x}_t, \mathbf{R}_t(\boldsymbol{\theta}_t)),$$
$$\mathbf{x}_t | \mathbf{x}_{t-1} \sim \mathcal{N}_n(\mathcal{M}_t(\mathbf{x}_{t-1}), \mathbf{Q}_t),$$
$$\theta_{t,l} \sim \mathcal{IG}(\kappa/2, \kappa/2), \quad l = 1, \ldots, m_t,$$

where $\mathbf{R}_t(\boldsymbol{\theta}) = \sigma_t^2 \operatorname{diag}(\theta_{t,1}, \ldots, \theta_{t,m_t})$, σ_t^2 is known, and $\boldsymbol{\theta}_t = (\theta_{t,1}, \ldots, \theta_{t,m_t})'$ follow independent inverse-gamma distributions. The transformation level is a point mass such that $\mathbf{z}_t = \mathbf{y}_t$.

Next, we consider non-continuous observations, such as binary data and right-skewed data with a point mass at zero (e.g., hourly rainfall amounts). The EnKF and its transformation- or quantile-based extensions (e.g., Anderson 2009; Amezcua and Van Leeuwen 2014) are particularly poorly suited for assimilation of such observations. In our HSSM framework, this

setting is handled using threshold observation models, in which f_t is determined by transformations of the form $z_{t,l} = g_t(y_{t,l}; \boldsymbol{\theta}_t)$, $l = 1, \ldots, m$, where $g_t(\cdot)$ is a deterministic function containing an indicator function:

Example 2 (*Threshold observation model for rainfall*). The observed data are modeled as $z_{t,l} = g(y_{t,l}; \boldsymbol{\theta}_t) = y_{t,l}^{\theta_t} \mathbb{1}_{\{y_{t,l} > 0\}}$, where $\mathbb{1}$ is the indicator function. Then $z_{t,l}$ follows a rainfall-type distribution, that is, a mixture of a positive, right-skewed distribution and a point mass at zero, for some $\theta_t > 1$ that determines the skewness (e.g., Sansó and Guenni 1999; Sigrist, Künsch, and Stahel 2012). The HSSM is given as

$$z_{t,l}|\mathbf{y}_t, \theta_t = y_{t,l}^{\theta_t} \mathbb{1}_{\{y_{t,l} > 0\}}, \quad l = 1, \ldots, m_t,$$
$$\mathbf{y}_t|\mathbf{x}_t \sim \mathcal{N}_{m_t}(\mathbf{H}_t\mathbf{x}_t, \sigma_t^2 \mathbf{I}_{m_t}),$$
$$\mathbf{x}_t|\mathbf{x}_{t-1} \sim \mathcal{N}_n(\mathcal{M}_t(\mathbf{x}_{t-1}), \mathbf{Q}_t),$$
$$\theta_t \sim p(\theta_t),$$

where the parameter θ_t is either known or assigned a prior distribution.

The threshold framework can also be used for *binary* data following a Bernoulli distribution with $z_{t,l}|\mathbf{x}_t, \boldsymbol{\theta}_t \sim \text{Bern}\left(\Phi((\mathbf{H}_t\mathbf{x}_t)_l/\sigma_t)\right)$, where $\Phi(\cdot)$ is the standard normal CDF. This is called a probit model for $\sigma_t = 1$. The transformation function is $z_{t,l} = g(y_{t,l}; \boldsymbol{\theta}_t) = \mathbb{1}_{\{y_{t,l} > 0\}}$, which corresponds to the rainfall model with parameter $\theta_t = 0$.

Example 3 (*Dynamic Poisson model*). For spatio-temporal count data, assume $f_t(\mathbf{z}_t|\mathbf{y}_t, \boldsymbol{\theta}_t) = \prod_{l=1}^{m_t} \text{Pois}(z_{t,l}|e^{y_{t,l}})$ and $\mathbf{R}_t = \sigma_t^2 \mathbf{I}_{m_t}$, where σ_t^2 accounts for overdispersion (Wikle 2002). The state follows a linear evolution, and the model includes six unknown time-varying parameters:

$$z_{t,l}|\mathbf{y}_t \sim \text{Pois}(\exp(y_{t,l})), \quad l = 1, \ldots, m_t,$$
$$\mathbf{y}_t|\mathbf{x}_t, \boldsymbol{\theta}_t \sim \mathcal{N}_{m_t}(\mathbf{H}_t\mathbf{x}_t, \sigma_t^2 \mathbf{I}_{m_t}),$$
$$\mathbf{x}_t|\mathbf{x}_{t-1}, \boldsymbol{\theta}_t \sim \mathcal{N}_n(\mathbf{M}(\boldsymbol{\gamma}_t)\mathbf{x}_{t-1}, \mathbf{Q}(\tau_t^2, \lambda_t)),$$
$$\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1} \sim \mathcal{N}_6(\boldsymbol{\theta}_{t-1}, .05^2\mathbf{I}_6),$$

where the transition matrix $\mathbf{M}(\boldsymbol{\gamma}_t)$ is tridiagonal with parameters $\boldsymbol{\gamma}_t = (\gamma_{1,t}, \gamma_{2,t}, \gamma_{3,t})$ and $\mathbf{Q}(\tau_t^2, \lambda_t)$ is a spatial covariance matrix with variance τ_t^2 and range λ_t. The unknown time-varying parameters $\boldsymbol{\theta}_t = (\gamma_{1,t}, \gamma_{2,t}, \gamma_{3,t}, \log \sigma_t, \log \tau_t, \log \lambda_t)'$ follow a random-walk evolution.

Example 4 (*Nonlinear evolution model with unknown static parameter*). The Lorenz-96 model is widely used in atmospheric science as an example of a nonlinear evolution operator with chaotic behavior (Lorenz 1996). Lorenz-96 mimics advection at $n = 40$ equally-spaced locations along a latitude circle on the globe. The time evolution is determined by

$$\frac{dx_{t,i}}{dt} = (x_{t,i+1} - x_{t,i-2})x_{t,i-1} - x_{t,i} + F, \quad i = 1, \ldots, 40,$$
$$(5)$$

with cyclic boundary conditions (i.e., $x_{t,38} = x_{t,-2}$, $x_{t,39} = x_{t,-1}$, $x_{t,41} = x_{t,1}$). Let $\text{Lor}_{8,0.2}$ denote the evolution operator implied by Equation (5) with forcing parameter $F = 8$ and time

step $\Delta t = 0.2$, which leads to strong nonlinearities and non-Gaussian forecast distributions. We observe data $\mathbf{z}_t = \mathbf{y}_t$ at each location so that $m = n = 40$. The HSSM model is

$$\mathbf{z}_t|\mathbf{y}_t \sim \delta_{\mathbf{y}_t}(\mathbf{z}_t),$$
$$\mathbf{y}_t|\mathbf{x}_t \sim \mathcal{N}_n(\mathbf{x}_t, \sigma^2\mathbf{I}_n),$$
$$\mathbf{x}_t|\mathbf{x}_{t-1}, \theta \sim \mathcal{N}_n(\mathcal{M}(\mathbf{x}_{t-1}; \theta), \mathbf{Q}),$$
$$\theta \sim \mathcal{N}(\mu_\theta, \sigma_\theta^2),$$

where σ^2 is known, $\mathcal{M}_t(\mathbf{x}_{t-1}; \theta) = \theta \, \text{Lor}_{8,0.2}(\mathbf{x}_{t-1})$ is the nonlinear evolution operator, which is unavailable in closed form, θ is an unknown parameter, and $\mathbf{Q}$ is a known covariance matrix.

1.3. Extended Ensemble Kalman Techniques

The goal in this paper is to make Bayesian inference on the states $\mathbf{x}_t$ and parameters $\boldsymbol{\theta}_t$ for HSSMs (1)–(4). Let $\mathbf{z}_{1:t} = \{\mathbf{z}_1, \ldots, \mathbf{z}_t\}$ denote the data up to time t, and use similar notation for $\mathbf{x}, \mathbf{y}$ and $\boldsymbol{\theta}$. Let $p(x|z)$ denote the conditional distribution of some random variable x given z. We consider two types of inference. *Filtering* involves computing the filtering distribution $p(\mathbf{x}_t, \boldsymbol{\theta}_t|\mathbf{z}_{1:t})$ sequentially for each time $t = 1, 2, \ldots$. *Smoothing* requires calculating the joint posterior (i.e., smoothing) distribution $p(\mathbf{x}_{1:T}, \boldsymbol{\theta}_{1:T}|\mathbf{z}_{1:T})$ for fixed time T.

We propose a *simple general idea* for such inference in high-dimensional HSSMs: Take an existing Bayesian inference technique (e.g., a particle filter or a Gibbs sampler), but approximate the part of the algorithm that requires sampling from or integrating out the high-dimensional state vector $\mathbf{x}_t$ with an EnKF or with the related ensemble Kalman smoother (EnKS). This leads to a general class of *extended ensemble Kalman filters and smoothers*, which can be applied to high-dimensional HSSMs. The key is that when conditioning on $\mathbf{y}_t$ and $\boldsymbol{\theta}_t$, our HSSM reduces to a standard Gaussian SSM with potentially nonlinear evolution, for which the EnKF is often well suited. Table 1 summarizes these algorithms and puts them in the context of existing particle approaches or approaches for linear Gaussian SSMs.

Our proposed filtering and smoothing algorithms can be classified into *Ensemble Gibbs* and *Ensemble Marginal* methods. For Ensemble Gibbs schemes, the EnKF update is used within a Gibbs sampler to draw from the full conditional distribution of $\mathbf{x}$ given $\boldsymbol{\theta}$ and $\mathbf{y}$. For Ensemble Marginal schemes, the EnKF is used to integrate out *and* sample from the posterior distribution of the states $\mathbf{x}$ within a particle filter for the parameters $\boldsymbol{\theta}$, similar to a mixture KF (Chen and Liu 2000). Further connections to existing methods will be provided with each of our proposed algorithms.

Our new extended EnK techniques can be viewed from two perspectives. First, these methods greatly expand the class of problems to which the EnKF and EnKS can be applied. Second, our techniques allow for accurate, approximate inference in higher-dimensional models than standard statistical methods such as particle filters and Gibbs samplers.

In the EnKF literature, several approaches exist for estimating specific tuning parameters, such as inflation and localization parameters (e.g., Wang and Bishop 2003; Anderson 2007a). Throughout, we assume that these tuning parameters are

Table 1. Existing and our new (in bold) methods for filtering and smoothing tasks for known and unknown parameters.

Target distrib.	Dim. of $\boldsymbol{\theta}_t$	Filtering/smoothing methods		
		Cond. Gaussian	Particle	Ensemble
$p(\mathbf{x}_t\|\boldsymbol{\theta}_{1:t}, \mathbf{y}_{1:t}), \forall t$	($\boldsymbol{\theta}$ known)	Kalman filter	Particle filter	EnKF (Alg. 1)
$p(\mathbf{x}_{1:T}\|\boldsymbol{\theta}_{1:T}, \mathbf{y}_{1:T})$	($\boldsymbol{\theta}$ known)	FFBS	Particle smoother	EnKS (Alg. 2)
$p(\mathbf{x}_t, \boldsymbol{\theta}_t\|\mathbf{z}_{1:t}), \forall t$	high	RWFFBS	Online MCMC	**GEnKF (Alg. 3)**
	low	RBPF	EARBPF	**PEnKF (Alg. 4)**
	high	Gibbs with FFBS	Particle Gibbs	**GEnKS (Alg. 5)**
$p(\mathbf{x}_{1:T}, \boldsymbol{\theta}_{1:T}\|\mathbf{z}_{1:T})$	low	MH with KF	PMMH	**MHEnKS/PEnKS (Alg. 6)**

NOTES: The algorithms for high-dimensional $\boldsymbol{\theta}_t$ generally require its full-conditional distribution to be available in closed form. "Cond. Gaussian" refers to methods where the full conditional distribution for the states is Gaussian. The methods in the column "Particle" do not scale to large state dimension n. Abbreviations and main references: KF = Kalman filter (Kalman 1960); particle filter (Gordon, Salmond, and Smith 1993); EnKF = ensemble Kalman filter (Evensen 1994); FFBS = forward-filtering backward-sampling (Carter and Kohn 1994); particle smoother (Kitagawa 1996); EnKS = ensemble Kalman smoother (Evensen and van Leeuwen 2000); RWFFBS = rolling-window FFBS (Polson, Stroud, and Müller 2008); Online MCMC (Berzuini et al. 1997); RBPF = Rao-Blackwellized PF (Doucet et al. 2000a); EARBPF = exact approximate RBPF (Johansen, Whiteley, and Doucet 2012); particle Gibbs (Andrieu, Doucet, and Holenstein 2010); MH with KF = Metropolis-Hastings based on KF likelihood (Schweppe 1965); PMMH = particle marginal Metropolis–Hastings (Andrieu, Doucet, and Holenstein 2010).

known, and we focus instead on inference of model parameters that explicitly appear in the model (1)–(4). Some approaches for state and parameter estimation in highly nonlinear SSMs (Gu and Oliver 2007; Chen and Oliver 2012; Bocquet and Sakov 2013) generate approximate posterior samples from the filtering or smoothing distribution using iterative optimization approaches. Specifically, the ensemble randomized maximum likelihood method (Chen and Oliver 2012) recasts posterior simulation as an optimization problem, and uses optimization methods with successive linearization of the model to find the posterior mode. These methods are quite general and can be used for parameter estimation and smoothing inference via state augmentation.

As the EnKF is an approximate technique for inference on the state vector, our extended EnK techniques also provide approximate inference. However, in many complex problems, exact inference is impossible, and approximate techniques such as approximate Bayesian computation (ABC; Sisson, Fan, and Tanaka 2007; Marin et al. 2012), synthetic likelihood (e.g., Wood 2010), and integrated nested Laplace approximations (INLA; Rue, Martino, and Chopin 2009; Lindgren, Rue, and Lindström 2011) have generally gained in popularity. While so-called exact approximate approaches (e.g., Andrieu, Doucet, and Holenstein 2010; Johansen, Whiteley, and Doucet 2012) result in exact inference given infinite computational resources, we will show here numerically that, in finite computation time, our extended EnK techniques can outperform exact approximate methods even for moderately high dimensions.

1.4. Overview and Contributions

This article is organized as follows. In Section 2, we review some existing work on the EnKF and related issues from a statistical perspective. In Section 3, we provide preliminaries to lay the foundation for the development of our extended EnK techniques, including new insights on likelihood approximations in high-dimensional SSMs. In Section 4, we derive and present our extended ensemble Kalman filters and smoothers. In Section 5, we compare our methods to existing approaches using a number of simulated and real data examples. We find that our methods strongly outperform these existing approaches. Conclusions are given in Section 6.

2. Review of Ensemble Kalman Filter and Smoother

In this section, we briefly review the ensemble Kalman filter (EnKF) and its extensions. The EnKF is a sequential Monte Carlo algorithm that generates samples from the state filtering distribution for the SSM (2) and (3), where $\mathbf{y}_{1:t}$ and $\boldsymbol{\theta}_{1:t}$ are known. Thus, throughout this section, we assume $\mathbf{y}_{1:t}$ and $\boldsymbol{\theta}_{1:t}$ are known and we often suppress $\boldsymbol{\theta}_{1:t}$ from the conditioning set. Further, let $\mathcal{N}_n(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denote an n-variate normal density with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, evaluated at $\mathbf{x}$.

As mentioned in Section 1.2, we assume here that the evolution operator $\mathcal{M}_t$ is intractable or computationally expensive, and so all we are able to do with $\mathcal{M}_t$ is to "apply" it to a finite and typically rather small number of state vectors. Thus, sequential inference over time must be based on an *ensemble* of state vectors that represents (i.e., approximates) the relevant distributions at each time point.

More specifically, assume that the *filtering ensemble* at time $t - 1$ is given by $\mathbf{x}_{t-1|t-1}^{(1:N)} := \{\mathbf{x}_{t-1|t-1}^{(1)}, \ldots, \mathbf{x}_{t-1|t-1}^{(N)}\}$, meaning that the filtering distribution at time $t - 1$ is approximated as

$$p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) = \frac{1}{N} \sum_{j=1}^N \delta_{(\mathbf{x}_{t-1|t-1}^{(j)})}(\mathbf{x}_{t-1}), \qquad (6)$$

where δ is the Dirac delta (point mass) function. The *forecast ensemble* $\widetilde{\mathbf{x}}_{t|t-1}^{(1:N)} = \{\widetilde{\mathbf{x}}_{t|t-1}^{(1)}, \ldots, \widetilde{\mathbf{x}}_{t|t-1}^{(N)}\}$ is obtained by applying $\mathcal{M}_t$ to each member of the filtering ensemble

$$\widetilde{\mathbf{x}}_{t|t-1}^{(j)} = \mathcal{M}_t(\mathbf{x}_{t-1|t-1}^{(j)}), \qquad j = 1, \ldots, N. \qquad (7)$$

2.1. Updating the State

Based on new data $\mathbf{y}_t$ observed at time t, we would like to update the forecast ensemble in Equation (7) to obtain a filtering ensemble from the filtering distribution of the state vector, $p(\mathbf{x}_t|\mathbf{y}_{1:t})$.

2.1.1. Update With a Mixture Prior
Based on Equation (6), the prior distribution at time t would in principle be given by

$$p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1}) dP(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})$$
$$= \frac{1}{N} \sum_{j=1}^N \mathcal{N}_n(\mathbf{x}_t|\widetilde{\mathbf{x}}_{t|t-1}^{(j)}, \mathbf{Q}_t), \qquad (8)$$

where the integral is with respect to the measure induced by the distribution of $\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}$. However, based on the mixture prior
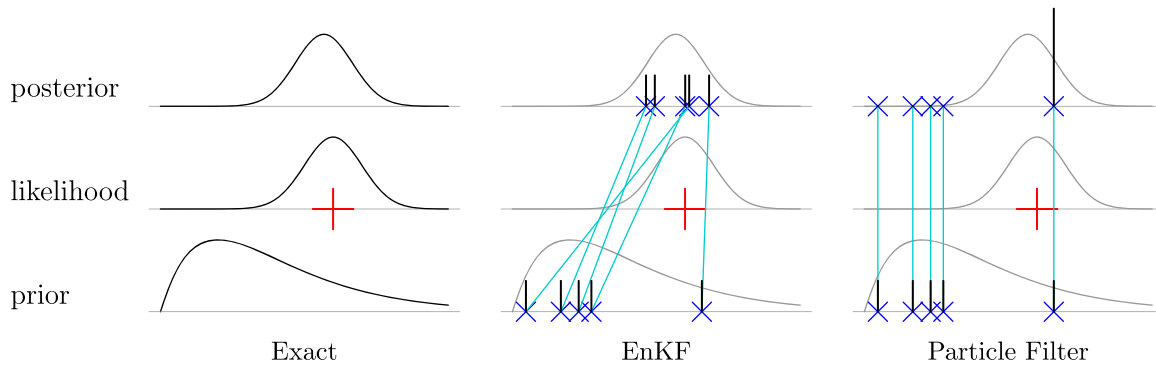
Figure 1. Comparison of updating methods in a simple one-dimensional example with a non-Gaussian prior distribution. The bars above the ensemble/particles are proportional to the weights. Both approximation methods start with the same, equally-weighted $N = 5$ prior samples. Even for this one-dimensional example, the importance weights degenerate for the particle filter, while the EnKF shifting update obtains a better representation of the posterior.

distribution in Equation (8), the filtering distribution updated using the new observation $\mathbf{y}_t$ would have the form

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \propto p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:t-1})$$
$$= \tfrac{1}{N}\sum_{j=1}^{N}\mathcal{N}_{m_t}(\mathbf{y}_t|\mathbf{H}_t\mathbf{x}_t,\mathbf{R}_t)\mathcal{N}_n(\mathbf{x}_t|\widetilde{\mathbf{x}}_{t|t-1}^{(j)},\mathbf{Q}_t)$$
$$\propto \sum_{j=1}^{N}\alpha_t^{(j)}p(\mathbf{x}_t|\mathbf{y}_t,\mathbf{x}_{t-1|t-1}^{(j)}), \qquad (9)$$

with the filtering or posterior weights

$$\alpha_t^{(j)} = p(\mathbf{x}_{t-1|t-1}^{(j)}|\mathbf{y}_{1:t}) \propto \mathcal{N}_{m_t}(\mathbf{y}_t|\mathbf{H}_t\widetilde{\mathbf{x}}_{t|t-1}^{(j)},\mathbf{H}_t\mathbf{Q}_t\mathbf{H}_t'+\mathbf{R}_t). \qquad (10)$$

This reweighting of the jth forecast ensemble member $\widetilde{\mathbf{x}}_{t|t-1}^{(j)}$ from prior weight $1/N$ to posterior weight $\alpha_t^{(j)}$ is essentially what happens in an importance sampler or in the update step of a particle filter that integrates out the innovation error with covariance $\mathbf{Q}_t$. This reweighting update is illustrated in a toy example in Figure 1 for the simple case $m_t = 1$, $\mathbf{H}_t = 1$, and $\mathbf{Q}_t = 0$. However, as shown in Snyder et al. (2008) by setting their observation error covariance matrix $\mathbf{R}$ to be $\mathbf{H}_t\mathbf{Q}_t\mathbf{H}_t'+\mathbf{R}_t$ in our notation, the particle weights *degenerate*, meaning that one mixture component would get all or nearly all of the weight, unless the ensemble size N increases exponentially with the effective dimension given by the variance of the particle loglikelihood (see Section 3.2.1 below). Thus, in our high-dimensional setting where only a small ensemble size N is computationally affordable, this update based on reweighting is problematic.

2.1.2. Update with a Gaussian Prior

In contrast to the particle filter, the EnKF update does not reweight but rather shifts the forecast ensemble. The form of the shift is motivated by the fact that if the evolution operator $\mathcal{M}_t$ is linear at each time t, then the true prior distribution will be Gaussian. In that case, it is natural to approximate the mixture of Gaussians in the prior distribution (8) by a single Gaussian distribution:

$$p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \mathcal{N}_n(\mathbf{x}_t|\widehat{\boldsymbol{\mu}}_{t|t-1},\widehat{\boldsymbol{\Sigma}}_{t|t-1}), \qquad (11)$$

where $\widehat{\boldsymbol{\mu}}_{t|t-1}$ and $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$ are estimates of the prior mean and covariance matrix (see Section 2.2) based on the forecast ensemble obtained in Equation (7). Given this Gaussian prior, the

posterior would also be Gaussian

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \propto p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:t-1})$$
$$= \mathcal{N}_{m_t}(\mathbf{y}_t|\mathbf{H}_t\mathbf{x}_t,\mathbf{R}_t)\mathcal{N}_n(\mathbf{x}_t|\widehat{\boldsymbol{\mu}}_{t|t-1},\widehat{\boldsymbol{\Sigma}}_{t|t-1})$$
$$\propto \mathcal{N}_n(\mathbf{x}_t|\widehat{\boldsymbol{\mu}}_{t|t},\widehat{\boldsymbol{\Sigma}}_{t|t}), \qquad (12)$$

where $\widehat{\boldsymbol{\mu}}_{t|t} = \widehat{\boldsymbol{\mu}}_{t|t-1} + \widehat{\mathbf{K}}_t(\mathbf{y}_t - \mathbf{H}_t\widehat{\boldsymbol{\mu}}_{t|t-1})$, $\widehat{\boldsymbol{\Sigma}}_{t|t} = (\mathbf{I} - \widehat{\mathbf{K}}_t\mathbf{H}_t)\widehat{\boldsymbol{\Sigma}}_{t|t-1}$, and

$$\widehat{\mathbf{K}}_t = \widehat{\boldsymbol{\Sigma}}_{t|t-1}\mathbf{H}_t'(\mathbf{H}_t\widehat{\boldsymbol{\Sigma}}_{t|t-1}\mathbf{H}_t'+\mathbf{R}_t)^{-1} \qquad (13)$$

is an estimate of the Kalman gain, $\mathbf{K}_t$.

2.1.3. The Ensemble Kalman Filter Update

In the stochastic EnKF update, instead of calculating the posterior distribution in Equation (12) explicitly, a sample from this distribution is obtained via *shifting* the forecast ensemble members by essentially carrying out conditional simulation (e.g., Katzfuss, Stroud, and Wikle 2016). This leads to the following algorithm:

Algorithm 1: Ensemble Kalman filter (EnKF)

Start with an initial ensemble $\mathbf{x}_{0|0}^{(1)},\ldots,\mathbf{x}_{0|0}^{(N)} \sim \mathcal{N}(\boldsymbol{\mu}_{0|0},\boldsymbol{\Sigma}_{0|0})$. Then, at each time $t = 1,\ldots,T$, do the following for each $j = 1,\ldots,N$:

1. **Forecast Step:** Compute forecast and prior ensemble members, $\widetilde{\mathbf{x}}_{t|t-1}^{(j)} = \mathcal{M}_t(\mathbf{x}_{t-1|t-1}^{(j)})$ and $\mathbf{x}_{t|t-1}^{(j)} = \widetilde{\mathbf{x}}_{t|t-1}^{(j)} + \mathbf{w}_t^{(j)}$, respectively, where $\mathbf{w}_t^{(j)} \sim \mathcal{N}_n(\mathbf{0},\mathbf{Q}_t)$.

2. **Update Step:** Shift each ensemble member as

$$\mathbf{x}_{t|t}^{(j)} = \mathbf{x}_{t|t-1}^{(j)} + \widehat{\mathbf{K}}_t(\mathbf{y}_t - \widetilde{\mathbf{y}}_t^{(j)}) =: \mathcal{U}_t(\widetilde{\mathbf{x}}_{t|t-1}^{(j)}|\mathbf{y}_t,\widetilde{\mathbf{x}}_{t|t-1}^{(1:N)},\boldsymbol{\theta}_t), \qquad (14)$$

where $\widehat{\mathbf{K}}_t$ is defined in (13), and $\widetilde{\mathbf{y}}_t^{(j)} = \mathbf{H}_t\mathbf{x}_{t|t-1}^{(j)} + \mathbf{v}_t^{(j)}$ is a pseudo-observation with $\mathbf{v}_t^{(j)} \overset{iid}{\sim} \mathcal{N}(\mathbf{0},\mathbf{R}_t)$.

The EnKF shifting update $\mathcal{U}_t$ in Equation (14) is derived based on assuming $\mathcal{M}_t$ is linear and hence that the prior distribution is multivariate Gaussian. The forecast ensemble $\widetilde{\mathbf{x}}_{t|t-1}^{(1:N)}$

enters $\mathcal{U}_t$ through the estimated Kalman gain $\widehat{\mathbf{K}}_t$, and $\mathbf{H}_t$, $\mathbf{Q}_t$, and $\mathbf{R}_t$ are evaluated at the parameter value $\boldsymbol{\theta}_t$, which is currently still assumed to be known. The resulting filtering distribution converges to the true filtering distribution as $N \to \infty$, assuming the estimator $\boldsymbol{\Sigma}_{t|t-1}$ converges to the true forecast covariance (see Section 2.2 below). For finite N, in contrast to many other SMC methods, the EnKF update does not deteriorate for large n, as long as a reasonable estimate of $\mathbf{K}_t$ or $\boldsymbol{\Sigma}_{t|t-1}$ can be found. It has been shown repeatedly (e.g., Lei, Bickel, and Snyder 2010) that this shifting update also works surprisingly well for nonlinear evolution or non-Gaussian priors (see Figure 1 for an illustration), including operational, real-world applications (e.g., Houtekamer and Mitchell 2005; Bonavita, Torrisi, and Marcucci 2010; Houtekamer et al. 2014).

2.2. Estimation of Matrices

The EnKF requires estimates of the mean vector $\widehat{\boldsymbol{\mu}}_{t|t-1}$ and covariance matrix $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$ of the prior distribution in Equation (11). The prior covariance matrix is required for the EnKF update through the Kalman gain (13), and both the mean and covariance matrix are needed in the EnKF likelihood in Section 3.2.2. The prior mean can simply be estimated by the sample mean of the forecast ensemble; that is, $\widehat{\boldsymbol{\mu}}_{t|t-1} = \frac{1}{N} \sum_{j=1}^{N} \widetilde{\mathbf{x}}_{t|t-1}^{(j)}$.

Estimation of the $n \times n$ prior covariance matrix based on small ensembles is often challenging in applications with large state dimension n, which can lead to ensemble collapse (e.g., Sætrom and Omre 2013) similar to the degeneration of the particle filter. In the EnKF, this collapse is counteracted by variance inflation (e.g., Anderson 2007a), data-dimension-reduction via shrinkage regression (Sætrom and Omre 2011), or regularization via tapering. We focus here on *tapering* (e.g., Furrer, Genton, and Nychka 2006; Anderson 2007b; Furrer and Bengtsson 2007), which is a very powerful method to avoid rank deficiency and spurious correlations by assuming that dependence is expected to decrease with increasing spatial distance. More precisely, tapering refers to entry-wise multiplication with a sparse positive-definite correlation matrix defined based on the distances between the elements. This leads to the expression $\widehat{\boldsymbol{\Sigma}}_{t|t-1} = \widetilde{\boldsymbol{\Sigma}}_{t|t-1} + \mathbf{Q}_t$ with $\widetilde{\boldsymbol{\Sigma}}_{t|t-1} = \mathbf{C}_t \circ \mathcal{T}_t$, where $\mathbf{C}_t$ is the sample covariance matrix of $\{\widetilde{\mathbf{x}}_{t|t-1}^{(1)}, \ldots, \widetilde{\mathbf{x}}_{t|t-1}^{(N)}\}$, and $\mathcal{T}_t$ is the sparse correlation matrix. Other regularization methods for states that are spatially referenced (e.g., Ott et al. 2004; Hunt, Kostelich, and Szunyogh 2007) or not (e.g., Pourahmadi 2013) will not be discussed here, but could also be applied in some of our methods described later. Also note that $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$ enters the Kalman gain in Equation (13) through the $n \times m_t$ matrix $\widehat{\boldsymbol{\Sigma}}_{t|t-1} \mathbf{H}_t'$ and the $m_t \times m_t$ matrix $\mathbf{H}_t \widehat{\boldsymbol{\Sigma}}_{t|t-1} \mathbf{H}_t'$, and so the EnKF can be stable even for massive n, as long as m_t is not too large.

2.3. Ensemble Kalman Smoother

To approximate the smoothing distribution $p(\mathbf{x}_t|\mathbf{y}_{1:T})$ for some T, various extensions of the EnKF to an EnKS have been proposed (van Leeuwen and Evensen 1996; Evensen and van Leeuwen 2000; Khare et al. 2008; Stroud et al. 2010; Bocquet and Sakov 2014). The most widely used smoother is the EnKS algorithm of Evensen and van Leeuwen (2000). Like the EnKF,

the EnKS is a forward-only algorithm that is applicable even for intractable evolution operators $\mathcal{M}_t$. At each time t, it essentially carries out an EnKF update on the augmented state that includes the entire history, $\mathbf{x}_{1:t} = (\mathbf{x}_1', \ldots, \mathbf{x}_t')'$:

Algorithm 2: Ensemble Kalman smoother (EnKS)

Start with an initial ensemble $\mathbf{x}_{0|0}^{(1)}, \ldots, \mathbf{x}_{0|0}^{(N)} \sim \mathcal{N}(\boldsymbol{\mu}_{0|0}, \boldsymbol{\Sigma}_{0|0})$. Then, at each time $t = 1, \ldots, T$, do the following for each $j = 1, \ldots, N$:

1. **Forecast Step:** Compute $\mathbf{x}_{t|t-1}^{(j)} = \mathcal{M}_t(\mathbf{x}_{t-1|t-1}^{(j)}) + \mathbf{w}_t^{(j)}$, where $\mathbf{w}_t^{(j)} \sim \mathcal{N}_n(\mathbf{0}, \mathbf{Q}_t)$.

2. **Update Step:** For $l = 1, \ldots, t$ compute $\mathbf{x}_{l|t}^{(j)} = \mathbf{x}_{l|t-1}^{(j)} + \widehat{\mathbf{K}}_{l,t}(\mathbf{y}_t - \widetilde{\mathbf{y}}_t^{(j)})$ where $\widetilde{\mathbf{y}}_t^{(j)} = \mathbf{H}_t \mathbf{x}_{t|t-1}^{(j)} + \mathbf{v}_t^{(j)}$ is the pseudo-observation with $\mathbf{v}_t^{(j)} \sim \mathcal{N}_{m_t}(\mathbf{0}, \mathbf{R}_t)$, $\widehat{\mathbf{K}}_{l,t} = \widehat{\boldsymbol{\Sigma}}_{l,t|t-1} \mathbf{H}_t'(\mathbf{H}_t \widehat{\boldsymbol{\Sigma}}_{t,t|t-1} \mathbf{H}_t' + \mathbf{R}_t)^{-1}$, and $\widehat{\boldsymbol{\Sigma}}_{l,t|t-1}$ is a regularized version of the sample cross-covariance matrix of $\mathbf{x}_{l|t-1}^{(1:N)}$ and $\mathbf{x}_{t|t-1}^{(1:N)}$.

The resulting ensemble $\mathbf{x}_{t|T}^{(1:N)}$ is then an approximate sample from the smoothing distribution $p(\mathbf{x}_t|\mathbf{y}_{1:T})$, and more generally, $\mathbf{x}_{1:T|T}^{(1:N)}$ is an approximate sample from $p(\mathbf{x}_{1:T}|\mathbf{y}_{1:T})$.

Note that regularization often results in $\widehat{\boldsymbol{\Sigma}}_{l,t|t-1} = \mathbf{0}$ for $l = 1, \ldots, t - k$ for some lag k. The computations in Step 2 of Algorithm 2 then only have to be carried out for k lags $l = t - k + 1, \ldots, t$.

3. Preliminaries for Extended Ensemble Kalman Techniques

In the previous section, we assumed that the latent $\mathbf{y}_{1:t}$ and the parameters $\boldsymbol{\theta}_{1:t}$ are all known. To obtain the necessary background and supporting results for the extended EnK techniques to be introduced in Section 4, we now extend the approaches reviewed in Section 2 to the more general HSSMs of the form Equations (1)–(4) with non-Gaussian observations or unknown parameters.

Assume that we have a sample $\boldsymbol{\theta}_{1:t-1}^{(1)}, \ldots, \boldsymbol{\theta}_{1:t-1}^{(M)}$ from $p(\boldsymbol{\theta}_{1:t-1}|\mathbf{z}_{1:t-1})$, the filtering distribution of the parameters at time $t - 1$. Except for special cases (see Section 3.1 below), we now need an ensemble of states, $\mathbf{x}_{t-1|t-1}^{(i,1:N)}$ corresponding to each parameter sample $\boldsymbol{\theta}_{1:t-1}^{(i)}$; that is, we need an *ensemble of ensembles*. Specifically, assume that the joint filtering distribution of parameters and states at time $t - 1$ is given by

$$p(\boldsymbol{\theta}_{1:t-1}, \mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})$$
$$= \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} \delta_{(\boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{x}_{t-1|t-1}^{(i,j)})}(\boldsymbol{\theta}_{1:t-1}, \mathbf{x}_{t-1}),$$

the marginal filtering distribution of $\boldsymbol{\theta}_{1:t-1}$ is $p(\boldsymbol{\theta}_{1:t-1}|\mathbf{z}_{1:t-1}) = \frac{1}{M} \sum_{i=1}^{M} \delta_{(\boldsymbol{\theta}_{1:t-1}^{(i)})}(\boldsymbol{\theta}_{1:t-1})$, and the conditional filtering distribution of $\mathbf{x}_{t-1}$ given $\boldsymbol{\theta}_{1:t-1}^{(i)}$ is

$$p(\mathbf{x}_{t-1}|\boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1}) = \frac{1}{N} \sum_{j=1}^{N} \delta_{(\mathbf{x}_{t-1|t-1}^{(i,j)})}(\mathbf{x}_{t-1}).$$

Conditional on $\boldsymbol{\theta}_{1:t-1}^{(i)}$, the joint prior distribution at time t is then given by

$$p(\boldsymbol{\theta}_t, \mathbf{x}_t | \boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1}) = p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}^{(i)}) p(\mathbf{x}_t | \boldsymbol{\theta}_t, \boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1}).$$

Further, we have

$$
\begin{aligned}
&p(\mathbf{x}_t | \boldsymbol{\theta}_t, \boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1}) \\
&= \int p(\mathbf{x}_t | \mathbf{x}_{t-1}, \boldsymbol{\theta}_t, \boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1}) dP(\mathbf{x}_{t-1} | \boldsymbol{\theta}_t, \boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1}) \\
&= \tfrac{1}{N} \sum_{j=1}^{N} \mathcal{N}_n(\mathbf{x}_t | \widetilde{\mathbf{x}}_{t|t-1}^{(i,j)}, \mathbf{Q}_t) \approx \mathcal{N}_n(\mathbf{x}_t | \widehat{\boldsymbol{\mu}}_{t|t-1}^{(i)}, \widehat{\boldsymbol{\Sigma}}_{t|t-1}^{(i)}), \quad (15)
\end{aligned}
$$

where we have used the same EnKF approximation as in Equation (11). Note that $\widehat{\boldsymbol{\mu}}_{t|t-1}^{(i)}$ and $\widehat{\boldsymbol{\Sigma}}_{t|t-1}^{(i)}$ are obtained as described in Section 2.2 using $\mathbf{Q}_t = \mathbf{Q}_t(\boldsymbol{\theta}_t)$, but only based on the ith forecast ensemble $\widetilde{\mathbf{x}}_{t|t-1}^{(i,1:N)}$, where $\widetilde{\mathbf{x}}_{t|t-1}^{(i,j)} = \mathcal{M}_t(\mathbf{x}_{t-1|t-1}^{(i,j)}; \boldsymbol{\theta}_t)$, $j = 1, \ldots, N$.

Based on the approximation in Equation (15), the joint distribution of all variables at time t is

$$
\begin{aligned}
&p(\mathbf{z}_t, \mathbf{y}_t, \mathbf{x}_t, \boldsymbol{\theta}_t | \boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1}) \\
&= p(\mathbf{z}_t | \mathbf{y}_t, \boldsymbol{\theta}_t) p(\mathbf{y}_t | \mathbf{x}_t, \boldsymbol{\theta}_t) p(\mathbf{x}_t | \boldsymbol{\theta}_t, \boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1}) p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}^{(i)}) \\
&= f_t(\mathbf{z}_t | \mathbf{y}_t, \boldsymbol{\theta}_t) \mathcal{N}_{m_t}(\mathbf{y}_t | \mathbf{H}_t \mathbf{x}_t, \mathbf{R}_t) \mathcal{N}_n(\mathbf{x}_t | \widehat{\boldsymbol{\mu}}_{t|t-1}^{(i)}, \widehat{\boldsymbol{\Sigma}}_{t|t-1}^{(i)}) p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}^{(i)})
\end{aligned}
$$
$$(16)$$

where $\mathbf{H}_t = \mathbf{H}_t(\boldsymbol{\theta}_t)$ and $\mathbf{R}_t = \mathbf{R}_t(\boldsymbol{\theta}_t)$ may also depend on the parameters $\boldsymbol{\theta}_t$.

3.1. Forecast Independence

An important special case is given by forecast independence of states and parameters (Frei and Künsch 2012), where $p(\widetilde{\mathbf{x}}_t, \boldsymbol{\theta}_t | \mathbf{z}_{1:t-1}) = p(\widetilde{\mathbf{x}}_t | \mathbf{z}_{1:t-1}) p(\boldsymbol{\theta}_t | \mathbf{z}_{1:t-1})$. Forecast independence leads to enormous *simplifications* in the algorithms described later, because the prior distribution of the states can be calculated based on the entire ensemble of ensembles from the previous filtering distributions, instead of using only the corresponding ensemble as in Equation (15); that is,

$$
\begin{aligned}
p(\widetilde{\mathbf{x}}_t | \boldsymbol{\theta}_t, \mathbf{z}_{1:t-1}) &= p(\widetilde{\mathbf{x}}_t | \mathbf{z}_{1:t-1}) = \int p(\widetilde{\mathbf{x}}_t | \mathbf{x}_{t-1}) dP(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}) \\
&= \mathcal{N}_n(\widetilde{\mathbf{x}}_t | \widehat{\boldsymbol{\mu}}_{t|t-1}, \widetilde{\boldsymbol{\Sigma}}_{t|t-1}),
\end{aligned}
$$
$$(17)$$

where $\widehat{\boldsymbol{\mu}}_{t|t-1}$ and $\widetilde{\boldsymbol{\Sigma}}_{t|t-1}$ are now based on all ensemble members $\widetilde{\mathbf{x}}_{t|t-1}^{(1:M,1:N)}$. From this forecast distribution, we can then obtain the prior distribution by adding $\mathbf{Q}_t$ to the covariance matrix: $\widehat{\boldsymbol{\Sigma}}_{t|t-1} = \widetilde{\boldsymbol{\Sigma}}_{t|t-1} + \mathbf{Q}_t$. Because this means that we do not need a separate ensemble for each previous parameter value, we can set $N = 1$ and write the forecast ensemble as $\widetilde{\mathbf{x}}_{t|t-1}^{(1:M)}$. Also, because the forecast distribution does not depend on $\boldsymbol{\theta}_t$, we only have to propagate the ensemble once from time $t-1$ to t in the algorithms described later, which is especially important if $\mathcal{M}_t$ is computationally expensive.

Forecast independence can be proved asymptotically if the parameters $\boldsymbol{\theta}_t$ only appear in $\mathbf{R}_t$ (see Frei and Künsch 2012). It also arises exactly if the parameters do not appear in $\mathcal{M}_t$ and are temporally independent, $p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}) = p(\boldsymbol{\theta}_t)$, because then

$$
\begin{aligned}
p(\widetilde{\mathbf{x}}_t | \boldsymbol{\theta}_t, \mathbf{z}_{1:t-1}) &= \int p(\widetilde{\mathbf{x}}_t | \mathbf{x}_{t-1}, \boldsymbol{\theta}_t) dP(\mathbf{x}_{t-1} | \mathbf{z}_{1:t-1}) \\
&= p(\widetilde{\mathbf{x}}_t | \mathbf{z}_{1:t-1}) = \mathcal{N}_n(\widetilde{\mathbf{x}}_t | \widehat{\boldsymbol{\mu}}_{t|t-1}, \widetilde{\boldsymbol{\Sigma}}_{t|t-1}),
\end{aligned}
$$

as in Equation (17). Instead of the joint distribution in Equation (16), it can then be shown that

$$
\begin{aligned}
&p(\mathbf{z}_t, \mathbf{y}_t, \mathbf{x}_t, \boldsymbol{\theta}_t | \mathbf{z}_{1:t-1}) \\
&= p(\mathbf{z}_t | \mathbf{y}_t, \boldsymbol{\theta}_t) p(\mathbf{y}_t | \mathbf{x}_t, \boldsymbol{\theta}_t) p(\mathbf{x}_t | \mathbf{z}_t, \mathbf{z}_{1:t-1}) p(\boldsymbol{\theta}_t) \\
&= f_t(\mathbf{z}_t | \mathbf{y}_t, \boldsymbol{\theta}_t) \mathcal{N}_{m_t}(\mathbf{y}_t | \mathbf{H}_t \mathbf{x}_t, \mathbf{R}_t) \\
&\quad \times \mathcal{N}_n(\mathbf{x}_t | \widehat{\boldsymbol{\mu}}_{t|t-1}, \widetilde{\boldsymbol{\Sigma}}_{t|t-1} + \mathbf{Q}_t) p(\boldsymbol{\theta}_t). \quad (18)
\end{aligned}
$$

3.2. Approximation of the Likelihood for Known $\mathbf{y}_t$

It is also possible to obtain an approximation of the likelihood based on the EnKF. We first study likelihood approximations for the special case of known and observed $\mathbf{y}_t$ (i.e., the hierarchical SSM given by Equations (2)–(4)).

The distribution of all data up to time t is then given by

$$
\begin{aligned}
p(\mathbf{y}_{1:t} | \boldsymbol{\theta}_{1:t}) &= \prod_{k=1}^{t} p(\mathbf{y}_k | \boldsymbol{\theta}_{1:t}, \mathbf{y}_{1:k-1}) \\
&= \prod_{k=1}^{t} p(\mathbf{y}_k | \boldsymbol{\theta}_{1:k}, \mathbf{y}_{1:k-1}),
\end{aligned}
$$

where we have now made the conditioning on the parameters explicit. However, in the high-dimensional setting with potentially nonlinear evolution considered here, the filtered likelihood at time t, $\mathcal{L}_t(\boldsymbol{\theta}_{1:t}) := p(\mathbf{y}_t | \boldsymbol{\theta}_{1:t}, \mathbf{y}_{1:t-1})$, is not available in closed form and cannot be exactly evaluated. Instead, in any parameter-inference procedure, we need to estimate this likelihood for given fixed parameter values based on the forecast ensemble $\widetilde{\mathbf{x}}_{t|t-1}^{(1:N)}$. For the remainder of Section 3.2, we focus on a single time point t and assume that $\widetilde{\mathbf{x}}_{t|t-1}^{(1:N)}$ is a sample from the true forecast distribution, $p(\widetilde{\mathbf{x}}_t | \mathbf{y}_{1:t-1}, \boldsymbol{\theta}_{1:t})$.

For Bayesian sampling-based inference, the closer the likelihood estimator is to the true likelihood, the better the performance of the parameter-inference procedure. Specifically, unbiased estimators of the true likelihood lead to so-called *exact approximate* procedures that can be shown to sample from the correct target distribution; but the *variability* of the estimator is also crucial, in that the variance of the logarithm of the estimator of the likelihood determines the performance of the parameter inference for finite computation time (Andrieu and Roberts 2009). Variances smaller than 2 are often recommended for satisfactory performance (e.g., Doucet et al. 2015).

3.2.1. Particle Likelihood

If the mixture prior distribution in Equation (8) were used, the filtered likelihood at time t would also be a mixture of normals,

$$
\begin{aligned}
p(\mathbf{y}_t | \boldsymbol{\theta}_{1:t}, \mathbf{y}_{1:t-1}) &= \tfrac{1}{N} \sum_{j=1}^{N} \mathcal{N}_{m_t}(\mathbf{y}_t | \mathbf{H}_t \widetilde{\mathbf{x}}_{t|t-1}^{(j)}, \mathbf{H}_t \mathbf{Q}_t \mathbf{H}_t' + \mathbf{R}_t) \\
&=: \mathcal{L}_t^{\mathrm{P}}(\mathbf{y}_t | \boldsymbol{\theta}_t, \widetilde{\mathbf{x}}_{t|t-1}^{(1:N)}). \quad (19)
\end{aligned}
$$

Except for the fact that we are able to integrate out the innovation error with covariance $\mathbf{Q}_t$ here due to the linear Gaussian observations, $\mathcal{L}_t^{\mathrm{P}}$ is the likelihood approximation used in several inference procedures based on particle filters (e.g., Andrieu, Doucet, and Holenstein 2010; Johansen, Whiteley, and Doucet 2012) and in some approaches in the EnKF literature (e.g., Ueno and Nakamura 2014, 2016). Using the notation from Equation (10), we can write this likelihood as $\mathcal{L}_t^{\mathrm{P}}(\mathbf{y}_t | \boldsymbol{\theta}_t, \widetilde{\mathbf{x}}_{t|t-1}^{(1:N)}) \propto (1/N) \sum_{j=1}^{N} \alpha_t^{(j)}$. While the $\alpha_t^{(j)}$ and thus also the *particle*

(a) Distribution of likelihood ratios for $n = 1$

(b) Distribution of likelihood ratios for $n = 6$

(c) Trace plots for MH (thinned by factor 5)

(d) Necessary N to keep the variance of the loglikelihood below 2
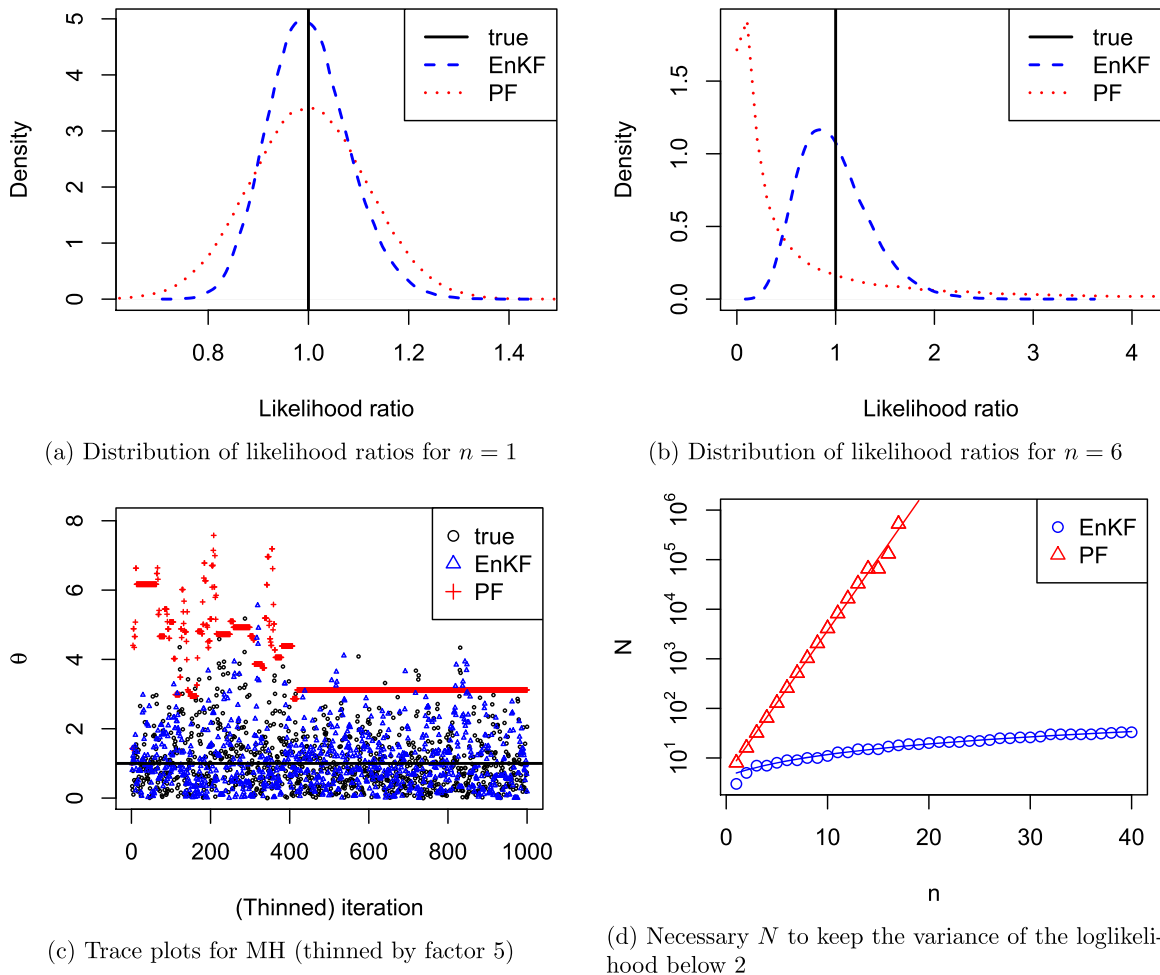
Figure 2. Comparison of the likelihoods discussed in Section 3.2 in the setting of Example 5 with $\kappa = 4$, and true parameter value $\theta = 1$. Panels (a) and (b): Distributions of EnKF and PF likelihood approximations divided by the true likelihood at $\theta = 1$ for $N = 50$ with fixed data **y** and varying forecast ensembles. Panel (c): Trace plot for "pseudo-marginal" Metropolis–Hastings (MH) algorithms targeting θ with a uniform prior on $\mathbb{R}^+$, $n = N = 50$, and a normal proposal distribution with standard deviation 0.8. Panel (d): Minimum N to keep the variance of the loglikelihood below 2, along with a least-squares line fitted on the original scale (for EnKF) and on a log-scale (for PF).

likelihood $\mathcal{L}_t^{\mathrm{P}}$ are unbiased for the true likelihood (Del Moral 2004), leading to exact approximate algorithms, the variance of the particle likelihood can be very large in high dimensions, resulting in algorithms with potentially poor performance in finite computation time. We now illustrate this using a simple example.

Example 5. For a single time point, consider the forecast distribution $\widetilde{\mathbf{x}} \sim \mathcal{N}_n(\mathbf{0}, \kappa \mathbf{I}_n)$, with model error covariance $\mathbf{Q} = \mathbf{0}$, noise covariance $\mathbf{R} = \theta \mathbf{I}_n$, observation matrix $\mathbf{H} = \mathbf{I}_n$, and thus $n = m$; that is, $\mathbf{y} \sim \mathcal{N}_n(\mathbf{0}, (\kappa + \theta)\mathbf{I}_n)$. The unknown parameter is θ.

In the setting of Example 5, Figure 2 shows the distribution of the particle likelihood $\mathcal{L}_t^{\mathrm{P}}$, which becomes increasingly skewed for increasing n. Further, it shows the trace plot for a "pseudo-marginal" Metropolis–Hastings (MH) sampler targeting θ for $n = 50$ using $\mathcal{L}_t^{\mathrm{P}}$, whose large variances cause the Markov chain to get "stuck" after a particularly large likelihood is obtained. Finally, we explore the ensemble size N necessary to keep the variance of the particle loglikelihood, which was averaged over 100 realizations of **y**, to below 2, and this number clearly scales *exponentially* with the dimension n. This result can also be shown analytically:

Proposition 1. In the setting of Example 5, we have $\mathrm{var}(\log \mathcal{L}_t^{\mathrm{P}} (\mathbf{y}|\theta, \widetilde{\mathbf{x}}^{(1:N)})) = \mathcal{O}(e^n/N)$ as $n, N \to \infty$, where the variance is evaluated with respect to $\widetilde{\mathbf{x}}^{(1:N)}$, and the $\widetilde{\mathbf{x}}^{(j)}$ are iid from the true forecast distribution.

All proofs are provided in Appendix A.

The results obtained here for a particle or importance-sampling likelihood at a single time point extend directly to a particle filter in a state-space setting. While a particle filter used to approximate a likelihood in a sequential setting will typically include a resampling step at every time point (e.g., Andrieu, Doucet, and Holenstein 2010), the resulting forecast or prior distribution at the next time point can be at most as good as the iid sampling from the true prior considered here, assuming that $\mathcal{M}_t$ is intractable, and so the particle-filter approximation to the likelihood will also degenerate in high dimensions.

3.2.2. EnKF Likelihood

Due to the problems of the PF likelihood in high dimensions, we use the EnKF approximation of the prior distribution in Equation (11) to obtain an *EnKF likelihood*

$$p(\mathbf{y}_t|\boldsymbol{\theta}_{1:t}, \mathbf{y}_{1:t-1}) = \int p(\mathbf{y}_t|\mathbf{x}_t, \boldsymbol{\theta}_{1:t}, \mathbf{y}_{1:t-1}) dP(\mathbf{x}_t|\boldsymbol{\theta}_{1:t}, \mathbf{y}_{1:t-1})$$

$$= \int \mathcal{N}_{m_t}(\mathbf{y}_t|\mathbf{H}_t\mathbf{x}_t, \mathbf{R}_t) \mathcal{N}_n(\mathbf{x}_t|\widehat{\boldsymbol{\mu}}_{t|t-1}, \widehat{\boldsymbol{\Sigma}}_{t|t-1}) d\mathbf{x}_t$$

$$= \mathcal{N}_{m_t}(\mathbf{y}_t|\mathbf{H}_t\widehat{\boldsymbol{\mu}}_{t|t-1}, \mathbf{H}_t\widehat{\boldsymbol{\Sigma}}_{t|t-1}\mathbf{H}_t' + \mathbf{R}_t)$$
$$=: \mathcal{L}_t^{\mathrm{E}}(\mathbf{y}_t|\boldsymbol{\theta}_t, \widetilde{\mathbf{x}}_{t|t-1}^{(1:N)}), \tag{20}$$

where $\mathbf{H}_t$, $\mathbf{R}_t$, and $\widehat{\boldsymbol{\mu}}_{t|t-1}$ and $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$ may depend on $\boldsymbol{\theta}_t$ through $\mathcal{M}_t$ and $\mathbf{Q}_t$.

The EnKF likelihood has been successfully used in several examples and applications (e.g., Mitchell and Houtekamer 2000; Stroud et al. 2010; Frei and Künsch 2012; Stroud, Katzfuss, and Wikle 2018). For linear SSMs, the true forecast distribution and likelihood are both exactly Gaussian, and so the EnKF likelihood converges to the true likelihood as $N \to \infty$, assuming no localization for $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$. For finite N, in contrast to the PF likelihood, the EnKF likelihood is a biased estimator of the true likelihood, and thus does not lead to exact approximate inference. Special care should be taken regarding parameters influenced by tapering, as, for example, a spatial-scale parameter in the innovation covariance $\mathbf{Q}_t$ might be overestimated to compensate for an overly restrictive taper in the forecast covariance $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$ (see Section 2.2). The estimated parameter is then useful for running an EnKF with the same taper settings, but it might not be the "true" spatial scale of the innovation covariance in the model (1)–(4).

However, more importantly, the variance of the EnKF (log)likelihood can be relatively small even in high dimensions, as long as a good regularized estimator $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$ can be found. This can be viewed as a manifestation of the well-known trade-off between bias and variance that is central to statistical learning with "big" data (e.g., Hastie, Tibshirani, and Friedman 2009). To illustrate this point, we consider the EnKF likelihood in the setting of Example 5 with a diagonal $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$. Figure 2 shows that the EnKF likelihood is less skewed, less variable, and thus closer to the true likelihood than the particle likelihood. This also leads to much better mixing for the MH sampler, and the resulting estimated posterior for θ (not shown) is nearly indistinguishable from the estimated posterior using the true likelihood. Figure 2(d) indicates that for the EnKF likelihood, N only has to increase *linearly* with the dimension n to keep the variance of the loglikelihood bounded. This can also be shown analytically:

Proposition 2. In the setting of Example 5, we have var(log $\mathcal{L}_t^{\mathrm{E}}$ $(\mathbf{y}|\boldsymbol{\theta}, \widetilde{\mathbf{x}}^{(1:N)})) = \mathcal{O}(n/N)$ as $n, N \to \infty$, where the variance is evaluated with respect to $\widetilde{\mathbf{x}}^{(1:N)}$, and the $\widetilde{\mathbf{x}}^{(j)}$ are iid from the true forecast distribution.

To explore whether this result holds more broadly than just in the independent case, we repeated the simulations in Figure 2(d) for a slightly modified version of Example 5 with a tridiagonal forecast covariance matrix $\boldsymbol{\Sigma}$. This produced a plot very similar to Figure 2(d).

If the true forecast distribution, and thus also the true likelihood, is non-Gaussian, the EnKF likelihood converges to the Dawid–Sebastiani scoring rule (Dawid and Sebastiani 1999) for $N \to \infty$, and so, from a frequentist perspective, parameter inference based on the EnKF likelihood at a single time point can be interpreted as approximate minimum contrast estimation (Pfanzagl 1969; Birgé and Massart 1993) under the Dawid–Sebastiani score. Due to the propriety of this scoring rule, the minimum-contrast estimator can be shown, using the proof

of Wald (1949), to be consistent under regularity conditions (see Gneiting and Katzfuss 2014, Sect. 3.4.2). From a Bayesian perspective, inference based on the EnKF likelihood essentially amounts to computing the approximate "Gibbs posterior" using the Dawid–Sebastiani score as the empirical risk function (Jiang and Tanner 2008; Li, Jiang, and Tanner 2013; Katzfuss and Bhattacharya 2015). Under certain conditions, most importantly that $\hat{\boldsymbol{\theta}}_t := \arg\max \mathcal{L}_t^{\mathrm{E}}(\mathbf{y}_t|\boldsymbol{\theta}_t, \widetilde{\mathbf{x}}_t^{(1:N)})$ is consistent and asymptotically normal as $m_t \to \infty$, it might be possible to scale and rotate posterior samples of $\boldsymbol{\theta}_t$ to reflect the true asymptotic uncertainty in the frequentist sense (Shaby 2014), but this will be pursued in future work.

3.3. Integrated Likelihood

We now extend the EnKF likelihood for known $\mathbf{y}_t$ from Section 3.2.2 to the full HSSM with the general transformation layer (1): $\mathbf{z}_t|\mathbf{y}_t, \boldsymbol{\theta}_t \sim f_t(\mathbf{z}_t|\mathbf{y}_t, \boldsymbol{\theta}_t)$. This *integrated EnKF likelihood* of the data $\mathbf{z}_t$ at time t, with $\mathbf{x}_t$ and $\mathbf{y}_t$ integrated out, is an important component of some of the algorithms described later. Using Equation (16), we have

$$p(\mathbf{z}_t|\boldsymbol{\theta}_t, \boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1}) = \int f_t(\mathbf{z}_t|\mathbf{y}_t, \boldsymbol{\theta}_t) \int \mathcal{N}_{m_t}(\mathbf{y}_t|\mathbf{H}_t\mathbf{x}_t, \mathbf{R}_t)$$
$$\times p(\mathbf{x}_t|\boldsymbol{\theta}_t, \boldsymbol{\theta}_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1})d\mathbf{x}_t d\mathbf{y}_t$$
$$= \int f_t(\mathbf{z}_t|\mathbf{y}_t, \boldsymbol{\theta}_t)\mathcal{L}_t^{\mathrm{E}}(\mathbf{y}_t|\boldsymbol{\theta}_t, \widetilde{\mathbf{x}}_{t|t-1}^{(i,1:N)})d\mathbf{y}_t$$
$$=: \mathcal{L}_t^{\mathrm{Z}}(\mathbf{z}_t|\boldsymbol{\theta}_t, \widetilde{\mathbf{x}}_{t|t-1}^{(i,1:N)}), \tag{21}$$

where $\mathcal{L}_t^{\mathrm{E}}$ is defined in (20), and $\mathbf{H}_t$, $\mathbf{R}_t$, and $\widehat{\boldsymbol{\mu}}_{t|t-1}^{(i)}$ and $\widehat{\boldsymbol{\Sigma}}_{t|t-1}^{(i)}$ may depend on $\boldsymbol{\theta}_t$.

Evaluating this likelihood requires carrying out an integral over the m_t-dimensional vector $\mathbf{y}_t$. Strategies for doing so depend on the choice of transformation distribution f_t. If the data $\mathbf{z}_t$ are Gaussian and the transformation is simply the identity, the integral degenerates and analytical integration is trivial. If it is possible to sample from $p(\mathbf{y}_t|\mathbf{z}_t, \mathbf{x}_t, \boldsymbol{\theta}_t)$, the integral can be approximated using these posterior samples (e.g., Chib and Jeliazkov 2001; Raftery et al. 2007). Finally, if $f_t(\mathbf{z}_t|\mathbf{y}_t, \boldsymbol{\theta}_t)$ is a continuous function in $\mathbf{y}_t$, a Laplace approximation (Tierney and Kadane 1986) can be used to approximate the integral. As we are already approximating $\mathbf{y}_t$ as a latent Gaussian field in $\mathcal{L}_t^{\mathrm{E}}(\mathbf{y}_t|\boldsymbol{\theta}_t, \widetilde{\mathbf{x}}_{t|t-1}^{(i,1:N)})$, the Laplace approximation of the integral should often result in negligible additional error, especially if the observations in $\mathbf{z}_t$ are independent given $\mathbf{y}_t$ and $\boldsymbol{\theta}_t$ (e.g., Rue, Martino, and Chopin 2009); see Section 5.2 for an example.

4. Extended Ensemble Kalman Techniques for Hierarchical SSMs

For HSSMs of the form Equations (1)–(4) with non-Gaussian observations or unknown parameters, we now propose extended EnK techniques. The general idea is to approximate the inference on the state vector in suitable existing Bayesian inference techniques using the EnKF or the EnKS.

4.1. Extended Ensemble Kalman Filters

We begin by discussing extended EnKFs that are suitable for filtering inference, which at each time t aims to find the filtering distribution $p(\mathbf{x}_t, \boldsymbol{\theta}_t | \mathbf{z}_{1:t})$ given the data $\mathbf{z}_{1:t}$ collected up to time t. Smoothing inference is discussed in Section 4.2.

4.1.1. Gibbs Ensemble Kalman Filter (GEnKF)

First, we consider a combination of an EnKF and a Gibbs sampler (Geman and Geman 1984; Gelfand and Smith 1990) for filtering inference in HSSMs. For this algorithm, we assume that the parameters do not appear in $\mathcal{M}_t$ and are temporally independent (i.e., $p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}) = p(\boldsymbol{\theta}_t)$). As noted in Section 3.1, this implies forecast independence, which enables inference based on a *single ensemble* $\widetilde{\mathbf{x}}_{t|t}^{(1:M)} = \{\widetilde{\mathbf{x}}_{t|t}^{(1)}, \ldots, \widetilde{\mathbf{x}}_{t|t}^{(M)}\}$ (i.e., $N = 1$). At each time t, our proposed Gibbs ensemble Kalman filter (GEnKF) iteratively samples $\mathbf{y}_t$, $\mathbf{x}_t$, and $\boldsymbol{\theta}_t$ from their respective full conditional distributions (FCDs), which are proportional to the joint distribution in (18). The state $\mathbf{x}_t$ is sampled by drawing from the forecast ensemble and using the EnKF update, which allows application of the GEnKF to high-dimensional and nonlinear SSMs.

Algorithm 3: Gibbs ensemble Kalman filter (GEnKF)

Starting with an initial ensemble $\{(\boldsymbol{\theta}_0^{(i)}, \mathbf{x}_{0|0}^{(i)}) : i = 1, \ldots, M\}$, where $(\boldsymbol{\theta}_0^{(i)}, \mathbf{x}_{0|0}^{(i)}) \sim p(\boldsymbol{\theta}_0, \mathbf{x}_0) = p_0(\boldsymbol{\theta}_0)\mathcal{N}_n(\mathbf{x}_0 | \boldsymbol{\mu}_{0|0}, \boldsymbol{\Sigma}_{0|0})$, do the following for each $t = 1, 2, \ldots$:

1. Obtain the forecast ensemble as $\widetilde{\mathbf{x}}_{t|t-1}^{(i)} = \mathcal{M}_t(\mathbf{x}_{t-1|t-1}^{(i)})$, for all $i = 1, \ldots, M$.
2. Find starting values for $\mathbf{y}_t^{(i)}$ and $\boldsymbol{\theta}_t^{(i)}$ for all $i = 1, \ldots, M$.
3. For $i = 1, \ldots, M$, iterate between the following steps until convergence:

   (a) Sample $\mathbf{x}_{t|t}^{(i)}$ from $p(\mathbf{x}_t | \mathbf{y}_t^{(i)}, \boldsymbol{\theta}_t^{(i)}, \mathbf{z}_{1:t})$ by sampling j uniformly from $\{1, \ldots, M\}$ and computing the EnKF update $\mathbf{x}_{t|t}^{(i)} = \mathcal{U}_t(\widetilde{\mathbf{x}}_{t|t-1}^{(j)} | \mathbf{y}_t^{(i)}, \widetilde{\mathbf{x}}_{t|t-1}^{(1:M)}, \boldsymbol{\theta}_t^{(i)})$ (see (14)).
   (b) Sample $\mathbf{y}_t^{(i)}$ from $p(\mathbf{y}_t | \mathbf{x}_{t|t}^{(i)}, \boldsymbol{\theta}_t^{(i)}, \mathbf{z}_{1:t}) \propto f_t(\mathbf{z}_t | \mathbf{y}_t, \boldsymbol{\theta}_t^{(i)})\mathcal{N}_{m_t}(\mathbf{y}_t | \mathbf{H}_t(\boldsymbol{\theta}_t^{(i)})\mathbf{x}_{t|t}^{(i)}, \mathbf{R}_t(\boldsymbol{\theta}_t^{(i)}))$.
   (c) Sample $\boldsymbol{\theta}_t^{(i)}$ from $p(\boldsymbol{\theta}_t | \mathbf{y}_t^{(i)}, \mathbf{x}_{t|t}^{(i)}, \mathbf{z}_{1:t}) \propto f_t(\mathbf{z}_t | \mathbf{y}_t^{(i)}, \boldsymbol{\theta}_t)\mathcal{N}_{m_t}(\mathbf{y}_t^{(i)} | \mathbf{H}_t(\boldsymbol{\theta}_t)\mathbf{x}_{t|t}^{(i)}, \mathbf{R}_t(\boldsymbol{\theta}_t)) \times \mathcal{N}_n(\mathbf{x}_t | \widehat{\boldsymbol{\mu}}_{t|t-1}, \widetilde{\boldsymbol{\Sigma}}_{t|t-1} + \mathbf{Q}_t(\boldsymbol{\theta}_t))p(\boldsymbol{\theta}_t)$.

After convergence, $(\mathbf{x}_{t|t}^{(i)}, \boldsymbol{\theta}_t^{(i)})$ is a joint sample from $p(\mathbf{x}_t, \boldsymbol{\theta}_t | \mathbf{z}_{1:t})$, and $\mathbf{x}_{t|t}^{(1:M)}$ is the filtering ensemble at time t.

Note that iterating through Step 3(a)–(c) of Algorithm 3 can be done completely independently (i.e., in parallel) for $i = 1, \ldots, M$. To sample j in Step 3(a), we use a systematic sample across $i = 1, \ldots, M$, so that each forecast ensemble member is chosen exactly once for each iteration of the Gibbs sampler. It can be shown under mild regularity conditions, that the closer the forecast ensemble is to multivariate normal, the closer the EnKF in Step 3(a) gets to sampling from the true FCD of $\mathbf{x}_t$ as M

increases, and hence the closer the resulting Markov chain will be to the Markov chain that would be obtained using an exact Gibbs sampler, in terms of total variation distance (Alquier et al. 2016). As the GEnKF requires sampling from the FCDs of $\mathbf{y}_t$ and $\boldsymbol{\theta}_t$ in (b) and (c), respectively, the algorithm is most efficient if the distributions $f_t(\mathbf{z}_t | \mathbf{y}_t, \boldsymbol{\theta}_t)$ and $p(\boldsymbol{\theta}_t)$ are such that these FCDs are available in closed form. Otherwise, Metropolis updates are necessary for these variables, which is generally only feasible if the number of parameters in $\boldsymbol{\theta}_t$ is not too large, and if the data in $\mathbf{z}_t$ are independent given $\mathbf{x}_t$.

For computational feasibility of Algorithm 3, it is crucial to find good starting values for $\mathbf{y}_t$ and $\boldsymbol{\theta}_t$, in order to minimize the required number of Gibbs iterations. In some applications, it is natural to use the values from the previous time point (i.e., set the initial values as $\mathbf{y}_t^{(i)} = \mathbf{y}_{t-1}^{(i)}$ and $\boldsymbol{\theta}_t^{(i)} = \boldsymbol{\theta}_{t-1}^{(i)}$). Another possibility is to choose the value of $\boldsymbol{\theta}_t$ that maximizes the integrated likelihood in (21).

As an example, for the model with heavy-tailed data in Example 1, the full-conditional distribution of $\boldsymbol{\theta}_t$ is a closed-form inverse-Gamma distribution, and so we can apply the following special case of the GEnKF in Algorithm 3:

Example 6 (Robust ensemble Kalman filter (REnKF)). Starting with an initial ensemble $\mathbf{x}_{0|0}^{(1:M)}$, where $\mathbf{x}_{0|0}^{(i)} \sim \mathcal{N}_n(\mathbf{x}_0 | \boldsymbol{\mu}_{0|0}, \boldsymbol{\Sigma}_{0|0})$, do the following for each $t = 1, 2, \ldots$:

1. Obtain the forecast ensemble as $\widetilde{\mathbf{x}}_{t|t-1}^{(i)} = \mathcal{M}_t(\mathbf{x}_{t-1|t-1}^{(i)})$, for all $i = 1, \ldots, M$.
2. Set $\mathbf{y}_t = \mathbf{z}_t$, and $\theta_{t,l}^{(i)} = 1$ for all $i = 1, \ldots, M$ and $l = 1, \ldots, m_t$.
3. For $i = 1, \ldots, M$, iterate between the following steps until convergence:

   (a) Sample $\mathbf{x}_{t|t}^{(i)}$ from $p(\mathbf{x}_t | \mathbf{y}_{1:t}, \boldsymbol{\theta}_t^{(i)})$ by sampling j uniformly from $\{1, \ldots, M\}$ and computing the EnKF update $\mathbf{x}_{t|t}^{(i)} = \mathcal{U}_t(\widetilde{\mathbf{x}}_{t|t-1}^{(j)} | \mathbf{y}_t^{(i)}, \widetilde{\mathbf{x}}_{t|t-1}^{(1:M)}, \boldsymbol{\theta}_t^{(i)})$ (see (14)).
   (b) Sample $\theta_{t,l}^{(i)} | \mathbf{y}_{1:t}, \mathbf{x}_{t|t}^{(i)} \overset{\text{ind}}{\sim} \mathcal{IG}\left(\frac{\kappa+1}{2}, \frac{\kappa}{2} + \left(\frac{y_{t,l} - (\mathbf{H}_t \mathbf{x}_{t|t}^{(i)})_l}{\sigma_t}\right)^2 / 2\right)$, $l = 1, \ldots, m_t$.

If κ is unknown with prior $p(\kappa)$, it could be sampled in a collapsed Gibbs sampling scheme prior to updating $\boldsymbol{\theta}_t$ by sampling from $p(\kappa | \mathbf{y}_t, \mathbf{x}_{t|t}^{(i)}) \propto p(\kappa) \prod_{l=1}^{m_t} t_\kappa((y_{t,l} - (\mathbf{H}_t \mathbf{x}_{t|t}^{(i)})_l)/\sigma_t)$.

Note that the iterative Huber-EnKF (Rao et al. 2017) seems to be similar in spirit to our robust EnKF, but it uses the Huber norm in place of the loss function implied by a t-distribution.

The GEnKF can also be applied to threshold models as in Example 2:

Example 7 (GEnKF for rainfall threshold model). Starting with an initial ensemble $\mathbf{x}_{0|0}^{(1:M)}$, where $\mathbf{x}_{0|0}^{(i)} \sim \mathcal{N}_n(\mathbf{x}_0 | \boldsymbol{\mu}_{0|0}, \boldsymbol{\Sigma}_{0|0})$, do the following for each $t = 1, 2, \ldots$:

1. Obtain the forecast ensemble as $\widetilde{\mathbf{x}}_{t|t-1}^{(i)} = \mathcal{M}_t(\mathbf{x}_{t-1|t-1}^{(i)})$, for all $i = 1, \ldots, M$.
2. Find starting values for $\mathbf{y}_t^{(i)}$ for all $i = 1, \ldots, M$.

3. For $i = 1, \ldots, M$, iterate between the following steps until convergence:

(a) Sample $\mathbf{x}_{t|t}^{(i)}$ from $p(\mathbf{x}_t|\mathbf{y}_{1:t}^{(i)})$ by sampling j uniformly from $\{1, \ldots, M\}$ and computing the EnKF update $\mathbf{x}_{t|t}^{(i)} = \mathcal{U}_t(\widetilde{\mathbf{x}}_{t|t-1}^{(j)}|\mathbf{y}_t^{(i)}, \widetilde{\mathbf{x}}_{t|t-1}^{(1:M)})$ (see (14)).

(b) Sample θ_t from $p(\theta_t|\mathbf{z}_t, \mathbf{x}_t) \propto p(\theta_t) \prod_{l:z_{t,l}>0} \mathcal{N}(z_{t,l}^{1/\theta_t}|(\mathbf{H}_t\mathbf{x}_t)_l, \sigma_t^2) z_{t,l}^{(1/\theta_t)-1}/\theta_t$.

(c) For $l = 1, \ldots, m_t$, sample $y_{t,l}$ from
$$p(y_{t,l}|z_{t,l}, \mathbf{x}_t, \theta_t) = \begin{cases} \delta_{z_{t,l}^{1/\theta_t}}(y_{t,l}), & z_{t,l} > 0, \\ \mathcal{N}^-((\mathbf{H}_t\mathbf{x}_t^{(i)})_l, \sigma_t^2), & z_{t,l} = 0, \end{cases}$$
where $\mathcal{N}^-$ denotes a truncated normal distribution on the negative real line.

A Gibbs ensemble Kalman filter could in principle also be applied if forecast independence does not hold. This is difficult, however, because $p(\theta_t, \mathbf{x}_t|\mathbf{z}_{1:t}) = \int p(\theta_t, \mathbf{x}_t|\theta_{t-1}, \mathbf{z}_{1:t}) dP(\theta_{t-1}|\mathbf{z}_{1:t})$, where it is straightforward to sample from $p(\theta_t, \mathbf{x}_t|\theta_{t-1}, \mathbf{z}_{1:t})$ using a Gibbs sampler similar to above, but it is highly challenging to compute the probability or weight $P(\theta_{t-1} = \theta_{t-1}^{(i)}|\mathbf{z}_{1:t})$, which will also degenerate in high dimensions. Thus, we will not pursue this general case here.

Perhaps the approaches in the literature most closely related to our GEnKF are Myrseth and Omre (2010) and Tsyrulnikov and Rakitko (2017), who model the forecast covariance matrix as random with an inverse-Wishart prior. This matrix is not a model parameter, which makes Bayesian inference difficult. Myrseth and Omre (2010) proposed a Gibbs-like algorithm with a single iteration, but they do not take the data into account when calculating the posterior of the forecast covariance. Tsyrulnikov and Rakitko (2017) seem to ignore the fact that forecast independence does not hold in their model.

4.1.2. Particle Ensemble Kalman Filter (PEnKF)
For high-dimensional SSMs, particle filters typically degenerate (e.g., Doucet, De Freitas, and Gordon 2001; Snyder et al. 2008). While Rao-Blackwellized particle filters (RBPFs; Doucet et al. 2000a) can scale to higher dimensions by integrating out part of the state analytically, for example using a Kalman filter (Chen and Liu 2000), the integral is generally not available for HSSMs with nonlinear evolution or non-Gaussian data. An exact approximation of RBPFs proposed in Johansen, Whiteley, and Doucet (2012) uses "local" particle filters to carry out the integration, thus employing the particle likelihood from Section 3.2.1, which was shown to scale poorly as well.

As an alternative, we propose here a *particle ensemble Kalman filter (PEnKF)*, which uses the EnKF likelihood from Sections 3.2.2 and 3.3 to approximately integrate out the high-dimensional state. The PEnKF is suitable for HSSMs for which forecast independence does not hold, or for which efficient sampling from the FCDs in Step 3 of Algorithm 3 is not possible. As a particle filter is applied to θ_t, the PEnKF will work best if the number of unknown parameters is not too high. Although this algorithm does not require forecast independence for the parameters as the GEnKF above, forecast independence does result in considerable simplifications (see below).

Algorithm 4: Particle ensemble Kalman filter (PEnKF)
Initialize the algorithm with an equally weighted ensemble $(\theta_0^{(i)}, \mathbf{x}_{0|0}^{(i,j)}) \sim p(\theta_0, \mathbf{x}_0) = p(\theta_0)\mathcal{N}_n(\mathbf{x}_0|\boldsymbol{\mu}_{0|0}, \boldsymbol{\Sigma}_{0|0})$ with $w_0^{(i)} = 1/M$, $i = 1, \ldots, M; j = 1, \ldots, N$. Then, for $t = 1, 2, \ldots$:

1. For $i = 1, \ldots, M$:

   (a) Sample a particle $\theta_t^{(i)}$ from a proposal distribution $q_t(\theta_t|\theta_{t-1}^{(i)})$.

   (b) Propagate the corresponding ensemble: $\widetilde{\mathbf{x}}_{t|t-1}^{(i,j)} = \mathcal{M}_t(\mathbf{x}_{t-1|t-1}^{(i,j)}; \theta_t^{(i)})$, $j = 1, \ldots, N$.

   (c) Calculate the particle weight:
   $w_t^{(i)} \propto w_{t-1}^{(i)} \mathcal{L}_t^Z(\mathbf{z}_t|\theta_t^{(i)}, \widetilde{\mathbf{x}}_{t|t-1}^{(i,1:N)}) p(\theta_t^{(i)}|\theta_{t-1}^{(i)})/ q_t(\theta_t^{(i)}|\theta_{t-1}^{(i)})$.

   (d) Generate $\mathbf{x}_{t|t}^{(i,j)}$ from $p(\mathbf{x}_t|\mathbf{z}_{1:t}, \theta_{1:t}^{(i)})$ for $j = 1, \ldots, N$ (see below).

2. The filtering distribution is given by
   $p(\theta_t, \mathbf{x}_t|\mathbf{z}_{1:t}) \approx \sum_{i=1}^M w_t^{(i)} \frac{1}{N} \sum_{j=1}^N \delta_{(\theta_t^{(i)}, \mathbf{x}_{t|t}^{(i,j)})}(\theta_t, \mathbf{x}_t)$.

3. If desired, resample $(\theta_t^{(i)}, \mathbf{x}_{t|t}^{(i,1:N)})$ to obtain an equally weighted sample (see, e.g., Douc, Cappé, and Moulines 2005, for a comparison of resampling schemes).

By noting that $p(\theta_{1:t}, \mathbf{z}_{1:t}) = p(\theta_t, \mathbf{z}_t|\theta_{1:t-1}, \mathbf{z}_{1:t-1})p(\theta_{1:t-1}, \mathbf{z}_{1:t-1})$, the expression for the weights in Step 1(c) is derived as follows:

$$w_t^{(i)} \propto \frac{w_{t-1}^{(i)}}{q_t(\theta_t^{(i)}|\theta_{t-1}^{(i)})} \frac{p(\theta_{1:t}^{(i)}, \mathbf{z}_{1:t})}{p(\theta_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1})}$$

$$= \frac{w_{t-1}^{(i)}}{q_t(\theta_t^{(i)}|\theta_{t-1}^{(i)})} p(\theta_t^{(i)}, \mathbf{z}_t|\theta_{1:t-1}^{(i)}, \mathbf{z}_{1:t-1}),$$

where $p(\theta_t, \mathbf{z}_t|\theta_{1:t-1}, \mathbf{z}_{1:t-1}) = p(\theta_t|\theta_{1:t-1}, \mathbf{z}_{1:t-1})$ $p(\mathbf{z}_t|\theta_{1:t}, \mathbf{z}_{1:t-1}) = p(\theta_t|\theta_{t-1})p(\mathbf{z}_t|\theta_{1:t}, \mathbf{z}_{1:t-1})$, and $p(\mathbf{z}_t|\theta_{1:t}^{(i)}, \mathbf{z}_{1:t-1})$ is approximated by the integrated likelihood in Equation (21).

Thus, the PEnKF is similar to the mixture KF (Chen and Liu 2000) and a Rao-Blackwellized particle filter, except that the high-dimensional state $\mathbf{x}$ is integrated out and sampled using the EnKF approximation. This again allows application to high-dimensional and nonlinear SSMs.

In the numerical examples in Section 5, for simplicity we set the proposal distributions equal to the prior distributions implied by the model: $q_t(\theta_t|\theta_{t-1}) = p(\theta_t|\theta_{t-1})$. Of course, it might be possible to adapt existing, more sophisticated strategies for choosing efficient proposal distributions to our setting.

For Step 2 in Algorithm 4, a sample from $p(\mathbf{x}_t|\mathbf{z}_{1:t}, \theta_{1:t}^{(i)})$ needs to be computed for each particle with significantly nonzero weight $w_t^{(i)}$. This sample can be obtained using a

special case of the Gibbs sampler in Algorithm 3 with $\boldsymbol{\theta}_t = \boldsymbol{\theta}_t^{(i)}$ held fixed (i.e., skipping Step 3(c)). In settings for which the Laplace approximation works well (see Section 3.3), this Gibbs sampler could consist of only a single iteration, in which $\mathbf{y}_t$ is first sampled from the Gaussian approximation to its FCD used in the Laplace approximation, and then $\mathbf{x}_t$ is sampled as in Step 3(a) of Algorithm 3. Here is an example of a PEnKF using this Laplace approach for the Poisson model in Example 3:

Example 8 (PEnKF for dynamic Poisson model). Initialize $(\boldsymbol{\theta}_0^{(i)}, \mathbf{x}_{0|0}^{(i,j)}) \sim p(\boldsymbol{\theta}_0)\mathcal{N}_n(\mathbf{x}_0|\boldsymbol{\mu}_{0|0}, \boldsymbol{\Sigma}_{0|0})$ and $w_0^{(i)} = 1/M$, $i = 1, \ldots, M; j = 1, \ldots, N$. Then, for $t = 1, 2, \ldots$, and each $i = 1, \ldots, M$:

1. Sample $\boldsymbol{\theta}_t^{(i)}$ from $\mathcal{N}_6(\boldsymbol{\theta}_{t-1}^{(i)}, .05^2\mathbf{I}_6)$.

2. Propagate the corresponding ensemble: $\widetilde{\mathbf{x}}_{t|t-1}^{(i,j)} = \mathbf{M}_t(\boldsymbol{\theta}_t^{(i)})\mathbf{x}_{t-1|t-1}^{(i,j)}, j = 1, \ldots, N$.

3. Using Newton–Raphson, find the mode $\boldsymbol{\nu}_t^{(i)}$, and the negative inverse Hessian $\boldsymbol{\Lambda}_t^{(i)}$ at the mode, of the function $\log g_t^{(i)}(\mathbf{y})$, where $g_t^{(i)}(\mathbf{y}) = \mathcal{L}_t^{\text{E}}(\mathbf{y}|\boldsymbol{\theta}_t^{(i)}, \widetilde{\mathbf{x}}_{t|t-1}^{(i,1:N)}) \prod_{l=1}^{m_t} \mathcal{P}ois(z_{t,l}|e^{y_l})$.

4. Calculate the particle weight $w_t^{(i)} \propto w_{t-1}^{(i)} \mathcal{L}_t^{\text{Z}}(\mathbf{z}_t|\boldsymbol{\theta}_t^{(i)}, \widetilde{\mathbf{x}}_{t|t-1}^{(i,1:N)})$, where $\mathcal{L}_t^{\text{Z}}(\mathbf{z}_t|\boldsymbol{\theta}_t^{(i)}, \widetilde{\mathbf{x}}_{t|t-1}^{(i,1:N)}) = g_t^{(i)}(\boldsymbol{\nu}_t^{(i)})/\mathcal{N}_{m_t}(\boldsymbol{\nu}_t^{(i)}|\boldsymbol{\nu}_t^{(i)}, \boldsymbol{\Lambda}_t^{(i)})$.

5. Ensemble update: $\mathbf{x}_{t|t}^{(i,j)} = \mathcal{U}_t(\widetilde{\mathbf{x}}_{t|t-1}^{(i,j)}|\mathbf{y}_t^{(i)}, \widetilde{\mathbf{x}}_{t|t-1}^{(i,1:N)}, \boldsymbol{\theta}_t^{(i)}), j = 1, \ldots, N$, where $\mathbf{y}_t^{(i)} \sim \mathcal{N}(\boldsymbol{\nu}_t^{(i)}, \boldsymbol{\Lambda}_t^{(i)})$.

6. Resample $(\boldsymbol{\theta}_t^{(i)}, \mathbf{x}_{t|t}^{(i,1:N)})$ with probability proportional to $w_t^{(i)}$; then set $w_t^{(i)} = 1/M$.

Algorithm 4 is, in principle, also applicable when $\boldsymbol{\theta}_t \equiv \boldsymbol{\theta}$ is constant over time (i.e., the prior distribution is a point mass at the previous value). However, in this case the proposal distribution also has to be a point mass, which means that over time there will be fewer and fewer unique particles. To avoid this, the particles can be resampled based on a kernel density estimate of the filtering distribution (Liu and West 2001; Frei and Künsch 2012), but then the state ensemble has to be propagated and updated again based on the newly sampled parameter values.

As described in Section 3.1, in the case of *forecast independence*, the likelihood can be computed based on the entire filtering ensemble from $t - 1$, and so we can set $N = 1$ and carry out Algorithm 4 with a *single ensemble* instead of an ensemble of ensembles. This important special case is described in Algorithm 1 of Frei and Künsch (2012). If $\mathcal{M}_t$ does not depend on $\boldsymbol{\theta}_t$ but is computationally expensive, it can then be advantageous to "oversample" the parameters, so that there are more parameter particles than state ensemble members.

4.2. Smoothing Using Extended Ensemble Kalman Techniques

We now discuss extended EnKS methods for HSSMs when smoothing inference is desired; that is, when interest is in $p(\mathbf{x}_{1:T}, \boldsymbol{\theta}_{1:T}|\mathbf{z}_{1:T})$ for some fixed time period $\{1, \ldots, T\}$.

4.2.1. Smoothing with High-Dimensional Parameters

If the parameters and $\mathbf{y}_t$ are high-dimensional but they can be efficiently sampled from their full-conditional distributions (FCDs), a Gibbs-EnKS can be applied:

Algorithm 5: Gibbs ensemble Kalman smoother (GEnKS)

1. Initialize $\mathbf{y}_{1:T}^{(0)}$ and $\boldsymbol{\theta}_{1:T}^{(0)}$.
2. Iterate between following steps for $i = 1, 2, \ldots$ until convergence:

   (a) Obtain samples $\mathbf{x}_{1:T|T}^{(i,1:N)}$ from $p(\mathbf{x}_{1:T}|\boldsymbol{\theta}_{1:T}^{(i-1)}, \mathbf{y}_{1:T}^{(i-1)})$ using the EnKS (Algorithm 2), and set $\mathbf{x}_{1:T|T}^{(i)} = \mathbf{x}_{1:T|T}^{(i,j)}$ for a j sampled uniformly at random from $\{1, \ldots, N\}$.

   (b) Draw a sample $\mathbf{y}_{1:T}^{(i)}$ from $p(\mathbf{y}_{1:T}|\mathbf{z}_{1:T}, \mathbf{x}_{1:T|T}^{(i)}, \boldsymbol{\theta}_{1:T}^{(i-1)})$.

   (c) Draw a sample $\boldsymbol{\theta}_{1:T}^{(i)}$ from $p(\boldsymbol{\theta}_{1:T}|\mathbf{x}_{1:T|T}^{(i)}, \mathbf{y}_{1:T}^{(i)}, \mathbf{z}_{1:T})$.

The GEnKS can be interpreted as an alternative version of the Gibbs-KF of Carter and Kohn (1994) or the particle-Gibbs algorithm of Andrieu, Doucet, and Holenstein (2010), where the Kalman or particle filter, respectively, is replaced by the EnKF. This allows the GEnKS to be applied to high-dimensional, nonlinear SSMs. In Section 5.3, we conduct a numerical comparison using the Lorenz-96 model from Example 4 between particle Gibbs and the following GEnKS:

Example 9 (GEnKS for Lorenz-96 with static parameter). Initialize $\theta^{(0)}$, set $\mathbf{y}_{1:T} = \mathbf{z}_{1:T}$, and then iterate between following steps for $i = 1, 2, \ldots$ until convergence:

1. Obtain samples $\mathbf{x}_{1:T|T}^{(i,1:N)}$ from $p(\mathbf{x}_{1:T}|\theta^{(i-1)}, \mathbf{y}_{1:T})$ using the EnKS (Algorithm 2), and set $\mathbf{x}_{1:T|T}^{(i)} = \mathbf{x}_{1:T|T}^{(i,j)}$ for a j sampled uniformly from $\{1, \ldots, N\}$.

2. With $\widetilde{\mathbf{x}}_{t|T}^{(i)} = \text{Lor}_{8,0.2}(\mathbf{x}_{t-1|T}^{(i)})$, draw a sample $\theta^{(i)}$ from

$$\mathcal{N}\left(\frac{\mu_\theta/\sigma_\theta^2 + \sum_{t=2}^T \widetilde{\mathbf{x}}_{t|T}^{(i)\prime}\mathbf{Q}_t^{-1}\mathbf{x}_{t-1|T}^{(i)}}{1/\sigma_\theta^2 + \sum_{t=2}^T \widetilde{\mathbf{x}}_{t|T}^{(i)\prime}\mathbf{Q}_t^{-1}\widetilde{\mathbf{x}}_{t|T}^{(i)}}, \frac{1}{1/\sigma_\theta^2 + \sum_{t=2}^T \widetilde{\mathbf{x}}_{t|T}^{(i)\prime}\mathbf{Q}_t^{-1}\widetilde{\mathbf{x}}_{t|T}^{(i)}} \right). \quad (22)$$

4.2.2. Smoothing with Low-Dimensional Parameters

If $\boldsymbol{\theta}_{1:T}$ cannot be sampled efficiently from its FCD in Step 1(c) of Algorithm 5 but $\boldsymbol{\theta}_t$ is low-dimensional, the following smoothing algorithms could be applied.

Static parameters. First, consider the case of a static (i.e., temporally constant) low-dimensional parameter vector. In this case, it is possible to apply a *Metropolis–Hastings-EnKS (MHEnKS)*. This algorithm consists of proposing a new parameter value, $\boldsymbol{\theta}^P$ say, and then accepting this value with a probability depending on the likelihood

$$p(\mathbf{z}_{1:T}|\boldsymbol{\theta}) = \prod_{t=1}^T p(\mathbf{z}_t|\mathbf{z}_{1:t-1}, \boldsymbol{\theta}) = \prod_{t=1}^T \mathcal{L}_t^{\text{Z}}(\mathbf{z}_t|\boldsymbol{\theta}, \widetilde{\mathbf{x}}_{t|t-1}^{(1:N)}) \quad (23)$$

for $\boldsymbol{\theta} = \boldsymbol{\theta}^P$, where the terms in the product on the right-hand side are the integrated likelihoods at time t in (21), which

can be obtained sequentially by carrying out an EnKF and computing the integrated likelihood at each time t. Then, we can obtain samples from $p(\mathbf{x}_{1:T}|\mathbf{z}_{1:T}, \boldsymbol{\theta}^{(i)})$ for each thinned posterior sample $\boldsymbol{\theta}^{(i)}$ using a special case of Algorithm 5, in which we hold $\boldsymbol{\theta}_t = \boldsymbol{\theta}^{(i)}$ fixed and skip Step 2(c). If the evolution is linear, it is also possible to reuse the filtering ensemble $\mathbf{x}_{T|T}^{(1:N)}$ already obtained in the EnKF by applying the EnKS from Stroud et al. (2010).

The MHEnKS can be interpreted as an alternative version of the particle marginal Metropolis–Hastings algorithm of Andrieu, Doucet, and Holenstein (2010), where the particle likelihood in Equation (19) is replaced by the EnKF likelihood in Equation (20). The MHEnKS should hence scale to higher state dimensions.

Time-varying parameters. If $\boldsymbol{\theta}_t$ varies over time, inference becomes more challenging, in that even if each $\boldsymbol{\theta}_t$ consists of a small number of elements, the combined vector of unknown parameters at all time points, $\boldsymbol{\theta}_{1:T}$, can still be rather high-dimensional if T is large. Hence, for a low-dimensional time-varying parameter vector, we propose the following particle-EnKS:

Algorithm 6: Particle ensemble Kalman smoother (PEnKS)

1. Obtain smoothing trajectories $\boldsymbol{\theta}_{1:T}^{(1:M)}$ and corresponding weights $w_T^{(1:M)}$ by running the PEnKF in Algorithm 4 up to time T, and saving and potentially resampling the trajectories $(\boldsymbol{\theta}_{1:t}^{(i)}, \mathbf{x}_{1:t|t}^{(i,1:N)})$ at each time t (Kitagawa 1996).

2. If desired, improve degeneracy by obtaining new state trajectories $\mathbf{x}_{1:T|T}^{(i,1:N)}$ from $p(\mathbf{x}_{1:T}|\mathbf{y}_{1:T}, \boldsymbol{\theta}_{1:T}^{(i)})$ for each particle trajectory $\boldsymbol{\theta}_{1:T}^{(i)}$ with significantly nonzero weight $w_T^{(i)}$, using a special case of Algorithm 5, in which we hold $\boldsymbol{\theta}_{1:T} = \boldsymbol{\theta}_{1:T}^{(i)}$ fixed and skip Step 2(c).

Note that this algorithm might perform poorly when T is large, because of degeneracy problems at "early" time points (e.g., Doucet, Godsill, and Andrieu 2000b). For Step 1, we can obtain the same simplifications in the case of forecast independence as for the "stand-alone" PEnKF described in Section 4.1.2. For Step 2, we can re-use state ensembles in the case of linear evolution using the EnKS procedure of Stroud et al. (2010).

4.2.3. A More General MCMC—EnKS

An extension of the Gibbs-EnKS in Algorithm 5, which we will call an MCMC–EnKS, is possible as long as $\boldsymbol{\theta}_t$ can be split into a set of parameters $\boldsymbol{\theta}_{1:T} = (\boldsymbol{\theta}_{1:T}^{(1)\prime}, \ldots, \boldsymbol{\theta}_{1:T}^{(J)\prime})'$, where each $\boldsymbol{\theta}_{1:T}^{(j)}$ is itself either low-dimensional or has a closed-form full-conditional distribution, and can be updated as in the Metropolis-Hastings-, Gibbs-, or particle-EnKS. This allows the fitting HSSMs with a variety of unknown parameters.

4.3. Properties of the Extended Ensemble Kalman Algorithms

4.3.1. Convergence

Proposition 3. If the evolution operators $\mathcal{M}_t$ are linear and $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$ is a consistent estimator of the forecast covariance matrix, the empirical distributions for the states and parameters produced by the PEnKF, GEnKF, MHEnKS, and PEnKS all converge to the true filtering or smoothing distributions under mild regularity conditions, as N, M, and the number of MCMC iterations all tend to infinity.

Our informal proof is derivative of existing results and provided in Appendix A. Note that $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$ is consistent if the tapering range in Section 2.2 tends to infinity as $N \to \infty$. The convergence in Proposition 3 does not hold exactly in the case of the GEnKF, in that the marginal filtering distribution of $\mathbf{x}_t$ integrated over $\boldsymbol{\theta}_t$ and $\mathbf{y}_t$ and the forecast distribution of $\mathbf{x}_{t+1}$ are not exactly Gaussian anymore, but the GEnKF treats them as such.

For SSMs with nonlinear evolution, the EnKF essentially converges to the best filter among those which update the state linearly based on the data at each time point (e.g., Le Gland, Monbet, and Tran 2011; Law, Tembine, and Tempone 2016). This is similar to the notions in the so-called linear Bayes estimation that considers linear estimators given only the first and second moments of the prior distribution and likelihood (e.g, Hartigan 1969; Goldstein and Wooff 2007). In addition, there is a close connection between approximate Bayes computation (ABC) and linear Bayes estimation (e.g., Nott et al. 2014) and the EnKF can be shown to be equivalent to special cases of ABC algorithms (Nott, Marshall, and Ngoc 2012). As argued in Section 3.2.2, the EnKF likelihood converges to the Dawid-Sebastiani score, which is a proper scoring rule. Thus, for nonlinear evolution the extended EnKF and EnKS techniques will not produce exact inference, but the resulting samples of the state and parameters can still provide useful approximations to the desired distributions in many settings.

For the high-dimensional applications of interest here, only small ensembles are computationally feasible, and so the algorithms' performance for *small N* and *M* is much more important than their asymptotic properties for N and M tending to infinity. Our algorithms will tend to perform well for HSSMs for which the embedded SSM (2)–(3) is well suited for inference using the EnKF and EnKS. This is the case for many real-world applications (e.g., Houtekamer and Mitchell 2005; Bonavita, Torrisi, and Marcucci 2010; Houtekamer et al. 2014). We have also seen in Section 3.2, that the EnKF approximation to the likelihood can be much less variable than the particle approximation, which should lead to better algorithms. Further numerical comparisons are provided in Section 5.

4.3.2. Computational Cost

The EnKF requires storing, propagating, and operating on N vectors (i.e., ensemble members) of length n, and the computational complexity of each EnKF update is $\mathcal{O}(nN^2)$ for most EnKF variants (e.g., Tippett et al. 2003). The computations at each time step are highly parallelizable.

For our extended EnKF algorithms, the EnKF generally has to be carried out several times. However, often only a small number of iterations or particles is necessary to obtain a reasonable approximation to the desired distribution (see Section 5 below). Note that in machine learning, often only a few or even a single update (called contrastive divergence) is used to sample from restricted Boltzman machine Markov random fields for training deep neural networks (Hinton 2002). Although this adds bias, empirical and some theoretical support suggests that it is an effective and efficient approach in the deep learning context (e.g., Bengio and Delalleau 2009). In addition, in the case of forecast independence for the GEnKF and PEnKF, only the update step has to be performed several times, while the forecast step only has to be carried out for a single ensemble. Thus, considering that the most expensive part of the EnKF is often the forward propagation using an expensive evolution model $\mathcal{M}_t$, the increased computational cost of the GEnKF or PEnKF relative to the EnKF can be negligible.

Thus, the EnKF and our proposed extensions are *scalable* to very high state dimensions n and large numbers of observations m_t. Parameters $\boldsymbol{\theta}_t$ that cannot be included in the state can also be high-dimensional if closed-form full-conditional distributions are available, so that our Gibbs ensemble Kalman techniques can be applied. However, operational, large-scale implementation for millions of dimensions or more is a significant undertaking. While efficient and parallel existing EnKF implementations such as the data assimilation research testbed, or DART (Anderson et al. 2009), should be suitable for the proposed extensions, the application of EnKF in operational, large-scale environments typically requires serious high-performance-computing expertise and extensive tuning of inflation and localization parameters.

5. Examples and Numerical Comparisons

The extended EnK methods described in Section 4 are very general and can be applied in many situations. We now present numerical comparisons for the examples in Section 1.2.1.

5.1. Simulation Study for Non-Gaussian Observations

First, we considered three non-Gaussian observation scenarios with temporally independent parameters with closed-form full conditionals, which can be handled using our GEnKF (Algorithm 3). In particular, we consider *heavy-tailed* observation noise from a t distribution with $\kappa = 2$ degrees of freedom from Example 1 (see Figure 3(a)), and the *rainfall* threshold model from Example 2 with the true $\kappa = 3$ assumed known (see Figure 3(b)) or unknown.

We implemented the special cases of the GEnKF presented in Examples 6 and 7 for the heavy-tailed and rainfall data, respectively. In the case of known θ_t for the rainfall data, we simply skipped Step 3(b) in Example 7. The GEnKFs were compared to a standard sequential importance resampling (SIR) particle filter, and to the true posterior distribution obtained using an MCMC that was run for 1000 iterations after 500 burn-in iterations. We also considered a commonly used extension of the EnKF update (Anderson 2003): For an observation equation

of the form $\mathbf{y}_t = \mathcal{H}_t(\mathbf{x}_t, \mathbf{v}_t)$, the pseudo-observation in (14) is then replaced by $\widetilde{\mathbf{y}}_t^{(j)} = \mathcal{H}_t(\mathbf{x}_{t|t-1}^{(j)}, \mathbf{v}_t^{(j)})$, and the Kalman gain is estimated as $\widehat{\mathbf{K}}_t = \widehat{\mathbf{C}}_{xy}\widehat{\mathbf{C}}_{yy}^{-1}$, where $\widehat{\mathbf{C}}_{yy}$ and $\widehat{\mathbf{C}}_{xy}$ are regularized versions of the empirical covariance matrix of $\widetilde{\mathbf{y}}_t^{(1:N)}$ and the cross-covariance matrix between $\mathbf{x}_{t|t-1}^{(1:N)}$ and $\widetilde{\mathbf{y}}_t^{(1:N)}$, respectively. For the rainfall data with unknown θ, we used state augmentation. For the heavy-tailed data, we also implemented the robust Huber-EnKF proposed in Roh et al. (2013). The Huber-EnKF requires a cutoff or clipping value for each observation location, which we selected by sampling 1000 times from the true forecast distribution based on an update consisting of a single observation at the respective location.

Because non-Gaussian observations are not directly related to the evolution model and the forecast step, we focused on the update step at a single time point t. We simulated 100 true state vectors of size $n = 100$ on a one-dimensional spatial domain $[1, 100]$ with $m_t = 75$ randomly chosen observation locations. The prior of the state was taken to be a normal distribution with known true mean 0.2 (constant over space) and known true covariance matrix based on a powered exponential covariance function with power 1.8 and scale parameter 10, same for all three simulation scenarios. Also, we assumed $\sigma_t = 0.2$ to be known for all examples.

For the standard EnKF, the Huber-EnKF, and the particle filter (PF), we used the same prior ensemble of size $M = 100$, simulated from the true prior distribution. For our GEnKF we only used a subset of size $M = 30$ of this ensemble, and only one Gibbs iteration for rainfall with known θ, and three iterations for the other two scenarios. For the EnKF and GEnKF, we used the same Wendland taper function with tapering length 20.

The results are shown in Figure 3 and Table 2. We computed both the mean squared prediction error (MSPE), which evaluates the accuracy of the posterior mean as a point prediction, and the continuous ranked probability score (CRPS), which evaluates the quality of the entire predictive distribution, including its shape and spread (see, e.g., Gneiting and Katzfuss 2014). For the heavy-tailed observations, we compared the state filtering distribution for the different methods to the true state, whereas for rainfall data we considered predictions of the true rainfall, $x_{t,l}\mathbb{1}_{\{x_{t,l}>0\}}$.

The GEnKF outperformed all competitors in all three scenarios, and came relatively close to the true posterior distribution. This is especially remarkable considering the GEnKF consisted of at most three EnKF updates with ensemble size 30, while the competitors each used an ensemble of size 100. Hence, in our simulations, the GEnKF produced the most accurate results at the lowest computational expense.

In Figure 4, we further investigated the computational costs by comparing the computation time and MSPE for the update in the GEnKF and the particle filter for varying ensemble sizes between 5 and 100 and number of particles between 30 and 1 million. The results ignore the potentially significant cost of applying evolution operators, to the benefit of the particle filter. All results are averaged over 20 repetitions. While the GEnKF obtains close to optimal MSPE even for very low computation times, the particle filter produces highly nonoptimal MSPEs even for computation times that are several orders of magnitude
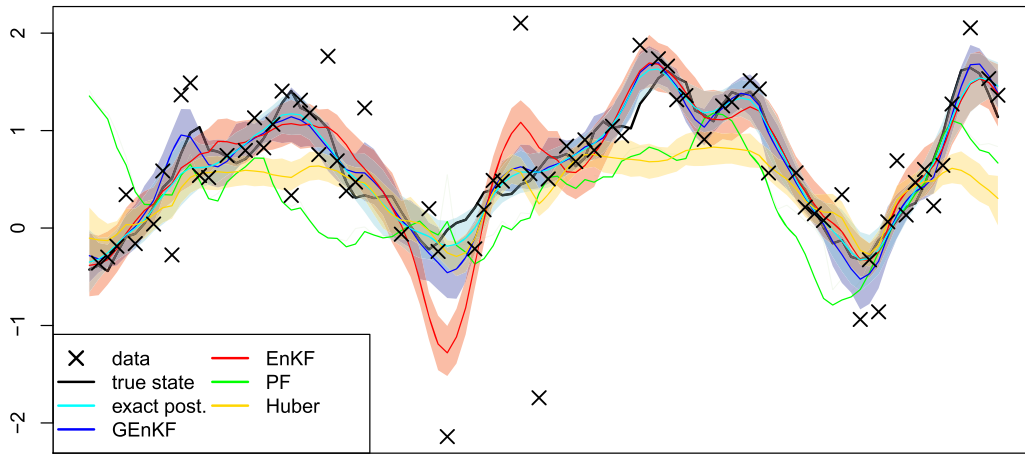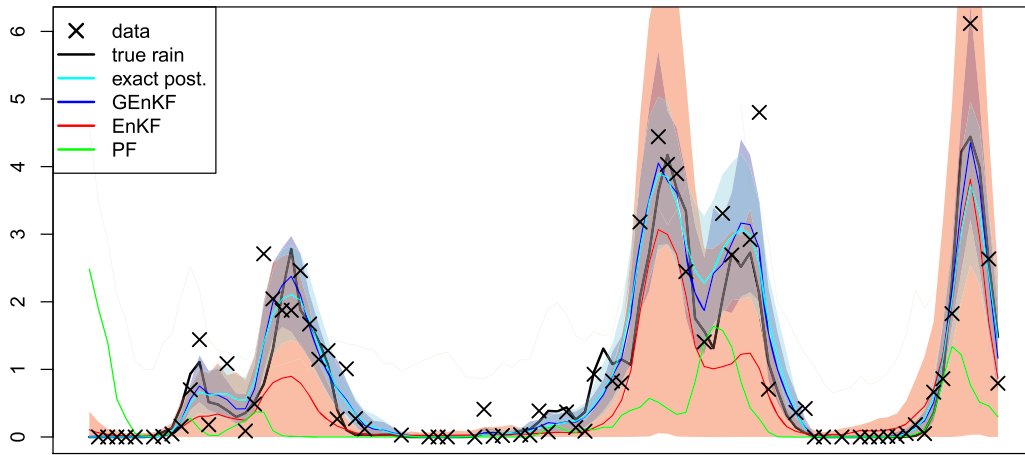
(a) Simulated heavy-tailed data with t_2 errors

(b) Simulated rainfall amounts with known $\theta = 3$

Figure 3. Simulated non-Gaussian data on a one-dimensional spatial domain, together with posterior means (solid lines) and pointwise posterior 80% credible intervals (shaded bands) as obtained by various methods.

Table 2. Simulation results for non-Gaussian observations: heavy-tailed data (Example 1) and rainfall data (Example 2).

	Heavy-tailed		Rainfall ($\theta = 3$)		Rainfall (θ unknown)	
	RMSPE	CRPS	RMSPE	CRPS	RMSPE	CRPS
exact	0.167	0.093	0.435	0.144	0.422	0.131
GEnKF	**0.185**	**0.103**	**0.452**	**0.152**	**0.939**	**0.220**
EnKF	0.288	0.149	26.714	7.427	>100	>100
PF	0.767	0.586	2.031	1.042	3.912	1.491
Huber	0.643	0.414				

NOTE: MSPE = mean squared prediction error; CRPS = continuous ranked probability score; exact = posterior distribution obtained using MCMC; PF = SIR particle filter; Huber = robust Huber-EnKF from Roh et al. (2013).

larger. The differences become even more pronounced for larger datasets.

5.2. Cloud Data

We also conducted filtering inference for a cloud-motion dataset shown in Figure 5a from Wikle (2002), representing cloud intensities (i.e., counts) at $n = 60$ locations along a spatial transect at $T = 80$ time points. The data roughly follow an overdispersed Poisson distribution, and so we assumed the model from Exam-

ple 3. We randomly set aside 10% of the observations as test data, so that $m_t \equiv 54$ and $\mathbf{H}_t$ was a subset of the identity. The observation locations were different for each time period. The evolution was taken to be linear, $\mathcal{M}_t(\mathbf{x}_{t-1}) = \mathbf{M}_t \mathbf{x}_{t-1}$, where $\mathbf{M}_t$ was a tridiagonal matrix with $\gamma_{1,t}$ on the main diagonal, and $\gamma_{2,t}$ and $\gamma_{3,t}$ on the first diagonals above and below, respectively. The evolution error was assumed to exhibit spatial dependence in the form of a Matérn covariance with smoothness 1.5, such that the (i, j)th element of $\mathbf{Q}_t$ is given by $(\mathbf{Q}_t)_{i,j} = \tau_t^2 \text{Mat}(|i - j|/\lambda_t)$, where $\text{Mat}(d) = (1 + \sqrt{3}d) \exp(-\sqrt{3}d)$. The unknown parameters were $\boldsymbol{\theta}_t := (\gamma_{1,t}, \gamma_{2,t}, \gamma_{3,t}, \log \sigma_t, \log \tau_t, \log \lambda_t)'$. For $t = 0$, we set $\boldsymbol{\theta}_0 = (0.3, 0.3, 0.3, \log(0.1), \log(1.5), \log(8))'$, $\boldsymbol{\mu}_{0|0} = (-2)\mathbf{1}_n$, and $(\boldsymbol{\Sigma}_{0|0})_{i,j} = 0.2 \text{Mat}(|i - j|/5)$. Note that our model addresses the potential extension to time-varying parameters suggested in Wikle (2002).

We considered two methods for filtering inference on the state and parameters: our PEnKF from Example 8, and an exact approximation of a Rao-Blackwellized particle filter (EARBPF; Johansen, Whiteley, and Doucet 2012). Forecast independence does not hold for all parameters in this example, and so we used $M = 50$ ensembles of size $N = 30$. For tapering of the forecast covariance matrix (see Section 2.2), we chose a Wendland taper with range 8. For the EARBPF, which is similar to the PEnKF
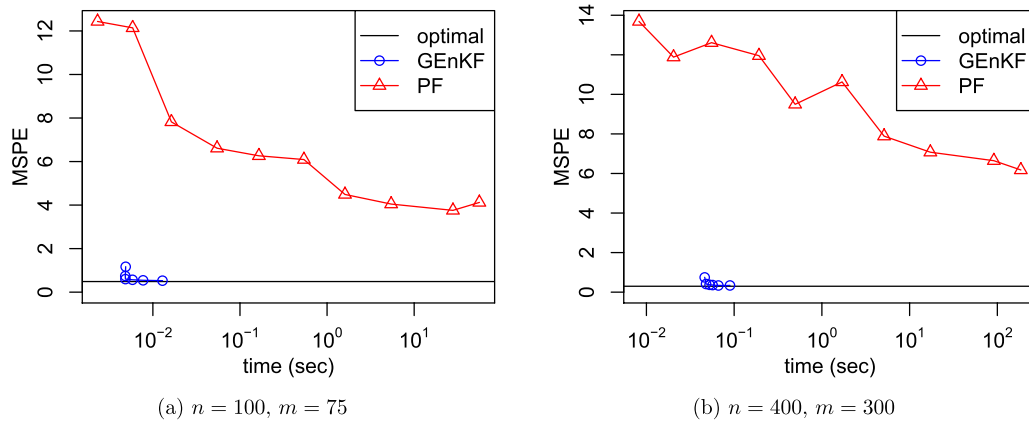
(a) $n = 100$, $m = 75$ (b) $n = 400$, $m = 300$

Figure 4. Mean squared prediction error (MSPE) versus time for GEnKF and particle filter (PF) updates in the simulated rain example for known $\kappa = 3$. Note that time on the *x*-axis is on a log scale.
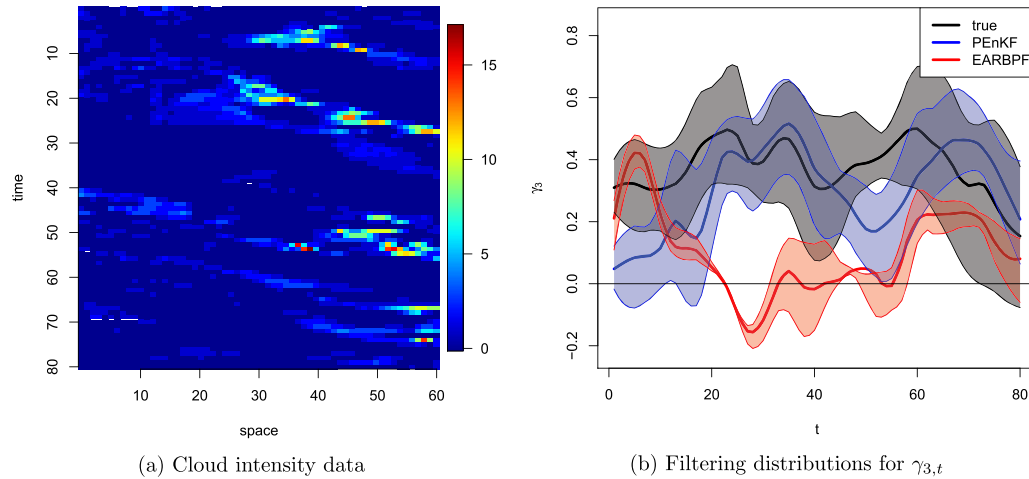

(a) Cloud intensity data (b) Filtering distributions for $\gamma_{3,t}$

Figure 5. The cloud data in Section 5.2 (Panel (a)), and filtering means and 95% credible intervals (slightly smoothed for clearness) over time for $\gamma_{3,t}$ (Panel (b)).

Table 3. For the cloud data in Section 5.2, mean squared prediction error (MSPE) and continuous ranked probability score (CRPS) for prediction of test data, and earth-mover's distance (EMD) to the true parameter posteriors.

	Predictions		EMD to true filtering distributions					
	MSPE	CRPS	γ_1	γ_2	γ_3	$\log \sigma$	$\log \tau$	$\log \lambda$
truth	0.73	0.25						
PEnKF	**0.75**	**0.25**	**0.25**	**0.16**	**0.19**	**0.70**	**0.23**	**0.32**
EARBPF	1.26	0.33	0.27	0.25	0.24	0.71	0.32	0.37

except that it uses local particle filters instead of the EnKF and the Laplace approximation to integrate out the state for each parameter particle, we considered one setting with a similar computation time as the PEnKF ($M = 50$ and $N = 30$), and one setting with a much larger computation time ($M = 500$ and $N = 10,000$) that should provide a close approximation to and will henceforth be referred to as the true filtering distribution.

The results, averaged over ten different initial ensembles $\mathbf{x}_{0|0}^{(1:M,1:N)}$, are shown in Table 3. In terms of filtering predictions of the test data, the PEnKF performed almost as well as the true filtering distributions, and considerably better than the EARBPF. The PEnKF approximation to the filtering distributions of the six parameters in $\boldsymbol{\theta}_t$ is also closer than the EARBPF to the true distributions in terms of earth-mover's

distance (computed using the R package by Urbanek and Rubner 2012) averaged over time. As an example, Figure 5(b) shows the filtering distributions for one initial ensemble over time of the parameter $\gamma_{3,t}$, which can be viewed as quantifying the amount of movement of the cloud intensities "one pixel to the right" from one time point to the next. This rightward drift, which is clearly present in the data (see Figure 5(a)), is picked up fairly quickly by the true and PEnKF filtering distributions, while the EARBPF approximation even implies a negative $\gamma_{3,t}$ between $t = 25$ and $t = 30$.

5.3. Smoothing Inference for the Lorenz-96 Model

Finally, we considered smoothing inference for simulated data based on the popular nonlinear Lorenz-96 model (Lorenz 1996) described in Example 4, with $\mathbf{Q} = 0.2\boldsymbol{\Sigma}_L$, where $\boldsymbol{\Sigma}_L$ is the covariance matrix obtained by a long run of the $\text{Lor}_{8,0.2}$ model. Starting with $\boldsymbol{\mu}_{1|0} = \mathbf{0}$ and $\boldsymbol{\Sigma}_{1|0} = \boldsymbol{\Sigma}_L$, we simulated data from the model for $t = 1, \ldots, T$ with $T = 10$, assuming $\theta \sim \mathcal{N}(\mu_\theta, \sigma_\theta^2)$ with $\mu_\theta = 0.8$ and $\sigma_\theta = 0.2$. The Lorenz-96 equations were solved numerically using an Euler scheme.

We simulated 100 such datasets, with the true θ drawn for each simulation from the prior. For each dataset, we ran three different methods to obtain approximations to the smoothing

(a) State at location 1 over time (i.e., $\mathbf{x}_{1:T,1}$)
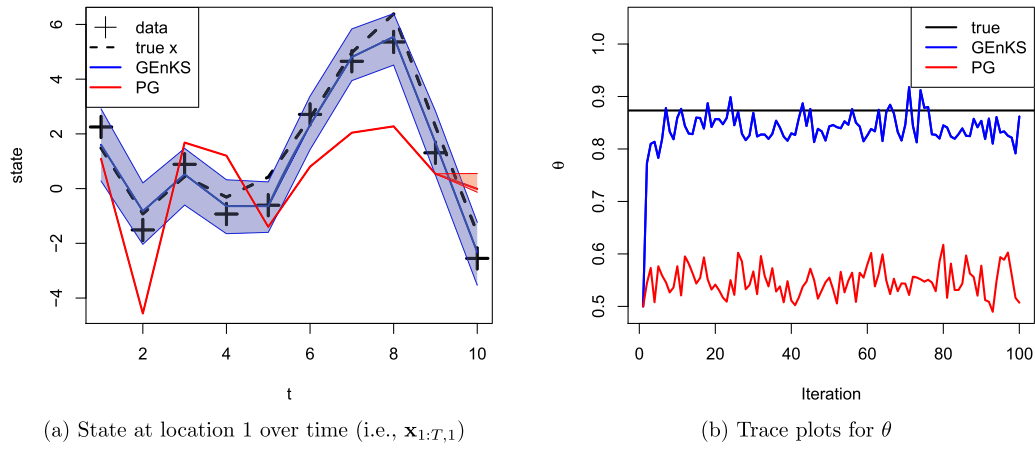
(b) Trace plots for θ

Figure 6. For one of the Lorenz-96 simulations in Section 5.3, true values and posterior distributions (i.e., posterior means and pointwise 80% prediction intervals) of the state at a single location (Panel (a)), and trace plots for θ (Panel (b))

Table 4. Mean squared prediction error (MSPE) and continuous ranked probability score (CRPS) for smoothing inference on the parameter θ and states $\mathbf{x}_{1:T}$ for the Lorenz-96 simulations in Section 5.3.

	Parameter θ		State $\mathbf{x}_{1:T}$	
	MSPE	CRPS	MSPE	CRPS
GEnKS	**0.002**	**0.024**	**0.710**	**0.478**
EnKS+SA	>100	>100	0.914	0.540
Particle Gibbs	0.111	0.262	12.380	2.495
Prior	0.042	0.118		

distribution $p(\theta, \mathbf{x}_{1:T}|\mathbf{y}_{1:T})$ using a starting value of $\theta = 0.5$. The first method considered was the EnKS in Algorithm 2 with $N = 1000$ and state augmentation; that is, θ was treated as part of the state vector $\mathbf{x}_t$ with artificial evolution distribution $\theta_t|\theta_{t-1} \sim \mathcal{N}(\theta_{t-1}, 0.1^2)$. We also considered the GEnKS in Example 9. For both EnKS algorithms with used a taper over time and space given by a Wendland correlation function with radius 3 and 8, respectively. Finally, we considered the particle Gibbs sampler (Andrieu, Doucet, and Holenstein 2010, Sect. 2.4.3), which uses the same distribution as the EnKS in Equation (22) to update θ, but uses a "conditional" particle filter to sample from the FCD of the state $\mathbf{x}_{1:T}$ given the data and θ. For both the GEnKS and the particle Gibbs sampler, we used an ensemble size of $N = 50$ and 100 Gibbs iterations, the first 20 of which were considered burn-in.

Due to the nonlinear evolution, it is not possible to obtain the true smoothing distribution of the states and parameter in any reasonable computation time. Hence, we compared the three methods using the same proper scoring rules as in Section 5.1, averaged over the 100 simulated datasets. The results are summarized in Table 4. Our GEnKS algorithm performed best both in terms of inference on the parameter and on the states. The state augmentation method failed, leading to very high (i.e., bad) scores for the posterior on θ. The particle Gibbs sampler produced worse inference on θ than simply using the prior distribution (i.e., completely ignoring any information in the data). Some of the results are also summarized in Figure 6, which shows the *collapse* of the particle Gibbs procedure for inference on $\mathbf{x}_{1:T}$ and its non-convergence in terms of inference on θ.

Note that while we considered θ to be the only unknown parameter here for clearness of exposition, our GEnKS should also work well for unknown parameters in $\mathbf{H}_t$ and scalar multiplicative parameters in $\mathbf{Q}_t$ and $\mathbf{R}_t$, in that closed-form FCDs are available in both of these cases if a normal and inverse-gamma priors, respectively, are assumed.

6. Conclusions

We have introduced a new class of ensemble filtering and smoothing algorithms for Bayesian inference in high-dimensional dynamic spatio-temporal models. The algorithms combine existing approaches from the state-space literature with ensemble Kalman methods from geophysics. The main advantage of our algorithms is that they scale well to high-dimensional models, and they allow for nonlinearity, non-Gaussian observations, and unknown parameters. In contrast, most existing state-space algorithms break down in high dimensions, and most ensemble Kalman approaches are poorly suited for non-Gaussian models with unknown parameters. We illustrated the proposed methods with four examples: two non-Gaussian models, a nonlinear geophysical system, and a real data example of cloud motion. We showed that our methods outperform the main competing approaches, including EnKFs, particle filters, and particle MCMC. Our methods also provide an approach for approximate Bayesian inference for large spatial data, in the special case of only $T = 1$ time point.

We have proposed two types of algorithms: *Ensemble Gibbs* and *Ensemble Marginal* methods. For Ensemble Gibbs methods, the EnKF is used for approximate simulation from the full conditional state distribution. In our examples, we show that these approximate Gibbs samplers can provide accurate state and parameter inference and converge within a small number of iterations. For Ensemble Marginalization methods, the EnKF is used to approximately integrate out the states and simulate from their full conditional posterior. In this context, we have studied the theoretical properties of EnKF-based likelihood estimates and shown that an ensemble size that is linear in the effective dimension is sufficient to bound the variance of the estimated loglikelihood. This is in contrast to particle-filter-based likelihood estimates, for which the required number of particles

grows exponentially in the effective dimension. We leave as future research further study on the theoretical properties of the proposed algorithms.

Statisticians now acknowledge that fast, approximate methods are necessary to analyze very large datasets. The EnKF, developed in the geophysics community, has been successfully applied for fast, approximate inference on models with millions of state variables. We believe that combining standard statistical approaches with these approximate but scalable EnKF methods provides solutions to problems that might be otherwise unsolvable.

Appendix A: Proofs

Proof of Proposition 1. For notational simplicity, we drop the $\sim$ on $\widetilde{\mathbf{x}}^{(j)}$, and we assume $\boldsymbol{\mu} = \mathbf{0}$ without loss of generality. First, note that $\sum_{i=1}^{n}(x_i^{(j)} - y_i)^2$ follows a noncentral χ^2-distribution, and so

$$\log \mathcal{L}^{\mathrm{P}}(\mathbf{y}|\boldsymbol{\theta}, \mathbf{x}^{(j)}) = -\frac{n}{2}\log(2\pi\theta) - \frac{1}{2\theta}\sum_{i=1}^{n}(x_i^{(j)} - y_i)^2$$
$$\xrightarrow{d} \mathcal{N}(na, nb),$$

where $a = -\frac{1}{2}(\log(2\pi\theta) + \kappa + \frac{1}{n}\sum_{i=1}^{n} y_i^2)$ and $b = 2\kappa(\kappa + \frac{2}{n}\sum_{i=1}^{n} y_i^2)$. Using properties of the lognormal distribution, we have

$$\mathcal{L}^{\mathrm{P}}(\mathbf{y}|\boldsymbol{\theta}, \mathbf{x}^{(1:N)}) = \frac{1}{N}\sum_{j=1}^{N}\exp(\log \mathcal{L}^{\mathrm{P}}(\mathbf{y}|\boldsymbol{\theta}, \mathbf{x}^{(j)}))$$
$$\xrightarrow{d} \mathcal{N}(e^{n(a+b/2)}, \frac{1}{N}e^{n2(a+b)}),$$

and finally, using the delta-method, we have the asymptotic variance

$$\operatorname{var}\log \mathcal{L}^{\mathrm{P}}(\mathbf{y}|\boldsymbol{\theta}, \mathbf{x}^{(1:N)}) \overset{a}{=} \frac{1}{N}e^{n2(a+b)}e^{-2n(a+b/2)}$$
$$= \frac{1}{N}e^{nb} = \frac{1}{N}e^{n2\kappa(\kappa + (2/n)\sum_{i=1}^{n} y_i^2)}$$
$$= \mathcal{O}(e^n/N).$$

Proof of Proposition 2. We drop the $\sim$ on $\widetilde{\mathbf{x}}^{(j)}$, and assume $\boldsymbol{\mu} = \mathbf{0}$ without loss of generality. Due to the diagonal tapering, the covariance matrix in the likelihood is a diagonal matrix with the ith element

$$\hat{\sigma}_i^2 = \frac{1}{N-1}\sum_{j=1}^{N}(x_i^{(j)} - \hat{\mu}_i)^2 + \theta \xrightarrow{d} \mathcal{N}(\sigma^2, 2\kappa^2/N),$$

where $\sigma^2 = \kappa + \theta$ and $\hat{\mu}_i = \frac{1}{N}\sum_{j=1}^{N} x_i^{(j)}$. Because $\operatorname{var}(\hat{\mu}_i^2) = \mathcal{O}(1/N^2)$, the variance of the mean estimator is negligible relative to that of $\hat{\sigma}_i^2$, and we regard $c_i = (y_i - \hat{\mu}_i)^2$ as a constant. Thus, by the delta method,

$$\operatorname{var}\left(\log \mathcal{L}^{\mathrm{E}}(y_i|\theta, x_i^{(1:N)})\right) = \operatorname{var}\left(\frac{1}{2}\log(\hat{\sigma}_i^2) + \frac{1}{2}\frac{c_i}{\hat{\sigma}_i^2}\right)$$
$$\overset{a}{=} \frac{1}{4}\frac{2\kappa^2}{N}\left(\frac{1}{\sigma^2} - \frac{c_i}{\sigma^4}\right)^2.$$

Finally, we have $\log \mathcal{L}^{\mathrm{E}}(\mathbf{y}|\theta, \mathbf{x}^{(1:N)}) = \sum_{i=1}^{n}\log \mathcal{L}^{\mathrm{E}}(y_i|\theta, x_i^{(1:N)})$, and so $\operatorname{var}(\log \mathcal{L}^{\mathrm{E}}(\mathbf{y}|\theta, \mathbf{x}^{(1:N)})) = \mathcal{O}(n/N)$. $\square$

Proof of Proposition 3. For linear Gaussian SSMs, the EnKF converges to the Kalman filter as $N \to \infty$ (e.g., Butala et al. 2008; Mandel, Cobb, and Beezley 2011), as long as $\widehat{\boldsymbol{\Sigma}}_{t|t-1}$ is a consistent estimator of the forecast covariance matrix. As the EnKS can be viewed as an EnKF

on an extended state-space, the same result holds for the EnKS. As stated in Section 3.2.2, the EnKF likelihood $\mathcal{L}_t^{\mathrm{E}}$ also converges to the true likelihood at each time t. Combining this with convergence results for MCMC (e.g., Robert and Casella 2004) and SMC (e.g., Doucet, De Freitas, and Gordon 2001) methods under mild regularity conditions, it is clear that the PEnKF, GEnKS, MHEnKS, and PEnKS all converge to the true filtering or smoothing distributions in a HSSM with linear evolution operators $\mathcal{M}_t$, as $N \to \infty$ and M or the number of MCMC iterations increase. $\square$

References

Alquier, P., Friel, N., Everitt, R., and Boland, A. (2016), "Noisy Monte Carlo: Convergence of Markov Chains with Approximate Transition Kernels," *Statistics and Computing*, 26, 29–47. [875]

Amezcua, J., and Van Leeuwen, P. J. (2014), "Gaussian Anamorphosis in the Analysis Step of the EnKF: A Joint State-variable/Observation Approach," *Tellus A*, 66, 1–18. [867]

Anderson, J., Hoar, T., Raeder, K., Liu, H., Collins, N., Torn, R., and Avellano, A. (2009), "The Data Assimilation Research Testbed: A Community Facility," *Bulletin of the American Meteorological Society*, 90, 1283–1296. [879]

Anderson, J. L. (2001), "An Ensemble Adjustment Kalman Filter for Data Assimilation," *Monthly Weather Review*, 129, 2884–2903. [867]

——— (2003), "A Local Least Squares Framework for Ensemble Filtering," *Monthly Weather Review*, 131, 634–642. [879]

——— (2007a), "An Adaptive Covariance Inflation Error Correction Algorithm for Ensemble Filters," *Tellus A*, 59, 210–224. [868,871]

——— (2007b), "Exploring the Need for Localization in Ensemble Data Assimilation Using an Hierarchical Ensemble Filter," *Physica D*, 230, 99–111. [871]

——— (2009),"Ensemble Kalman Filters for Large Geophysical Applications," *IEEE Control Systems Magazine*, 29, 66–82. [867]

——— (2010), "A Non-Gaussian Ensemble Filter Update for Data Assimilation," *Monthly Weather Review*, 138, 4186–4198. [867]

Andrews, D., and Mallows, C. (1974), "Scale Mixtures of Normality,". *Journal of the Royal Statistical Society*, Series B, 36, 99–102. [867]

Andrieu, C., Doucet, A., and Holenstein, R. (2010), "Particle Markov Chain Monte Carlo," *Journal of the Royal Statistical Society*, Series B, 72, 1–33. [869,872,873,877,878,882]

Andrieu, C., and Roberts, G. O. (2009), "The Pseudo-marginal Approach for Efficient Monte Carlo Computations," *Annals of Statistics*, 37, 697–725. [872]

Bengio, Y. and Delalleau, O. (2009), "Justifying and Generalizing Contrastive Divergence," *Neural Computation*, 21, 1601–1621. [879]

Berzuini, C., Best, N. G., Gilks, W. R., and Larizza, C. (1997), "Dynamic Conditional Independence Models and Markov Chain Monte Carlo Methods," *Journal of the American Statistical Association*, 92, 1403–1412. [869]

Birgé, L., and Massart, P. (1993), "Rates of Convergence for Minimum Contrast Estimators," *Probability Theory and Related Fields*, 97, 113–150. [874]

Bocquet, M., and Sakov, P. (2013), "Joint State and Parameter Estimation with an Iterative Ensemble Kalman Smoother," *Nonlinear Processes in Geophysics*, 20, 803–818. [869]

———— (2014), "An Iterative Ensemble Kalman Smoother," *Quarterly Journal of the Royal Meteorological Society*, 140, 1521–1535. [871]

Bonavita, M., Torrisi, L., and Marcucci, F. (2010), "Ensemble Data Assimilation with the CNMCA Regional Forecasting System," *Quarterly Journal of the Royal Meteorological Society*, 136, 132–145. [871,878]

Butala, M. D., Yun, J., Chen, Y., Frazin, R. A., and Kamalabadi, F. (2008), "Asymptotic Convergence of the Ensemble Kalman Filter," *2008 15th IEEE International Conference on Image Processing*, San Diego, CA, 825–828. [883]

Carlin, B. P., Polson, N. G., and Stoffer, D. S. (1992). "A Monte Carlo Approach to Nonnormal and Nonlinear State-space Modeling," *Journal of the American Statistical Association*, 87, 493–500. [866,867]

Carter, C., and Kohn, R. (1994), "On Gibbs Sampling for State Space Models," *Biometrika*, 81, 541–553. [866,867,869,877]

Chen, R., and Liu, J. S. (2000), "Mixture Kalman Filters," *Journal of the Royal Statistical Society*, Series B, 62, 493–508. [866,867,868,876]

Chen, Y., and Oliver, D. S. (2012), "Ensemble Randomized Maximum Likelihood Method as an Iterative Ensemble Smoother," *Mathematical Geosciences*, 44, 1–26. [869]

Chib, S., and Jeliazkov, I. (2001), "Marginal Likelihood from the Metropolis–Hastings Output," *Journal of the American Statistical Association*, 96, 270–281. [874]

Cressie, N., and Wikle, C. K. (2011), *Statistics for Spatio-Temporal Data*, Hoboken, NJ: Wiley. [867]

Dawid, A. P., and Sebastiani, P. (1999), "Coherent Dispersion Criteria for Optimal Experimental Design," *Annals of Statistics*, 65–81. [874]

Del Moral, P. (2004). *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*, New York: Springer. [873]

DelSole, T., and Yang, X. (2010). "State and Parameter Estimation in Stochastic Dynamical Models," *Physica D*, 239, 1781–1788. [867]

Douc, R., Cappé, O., and Moulines, E. (2005), "Comparison of Resampling Schemes for Particle Filtering," in *ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005*, pp. 64–69. [876]

Doucet, A., De Freitas, N., and Gordon, N. (2001), *Sequential Monte Carlo Methods in Practice*. New York: Springer. [866,876,883]

Doucet, A., de Freitas, N., Murphy, K., and Russell, S. (2000a). "Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks," in *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, pp. 176–183. [869,876]

Doucet, A., Godsill, S., and Andrieu, C. (2000b). "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Statistics and Computing*, 10(3):197–208. [878]

Doucet, A., Pitt, M. K., Deligiannidis, G., and Kohn, R. (2015), "Efficient Implementation of Markov Chain Monte Carlo When Using an Unbiased Likelihood Estimator," *Biometrika*, 102, 295–313. [872]

Evensen, G. (1994), "Sequential Data Assimilation With a Nonlinear Quasi-Geostrophic Model Using Monte Carlo Methods to Forecast Error Statistics," *Journal of Geophysical Research*, 99, 10143–10162. [866,869]

Evensen, G., and van Leeuwen, P. J. (2000), "An Ensemble Kalman Smoother for Nonlinear Dynamics," *Monthly Weather Review*, 128, 1852–1867. [869,871]

Frei, M., and Künsch, H. R. (2012), "Sequential State and Observation Noise Covariance Estimation Using Combined Ensemble Kalman and Particle Filters," *Monthly Weather Review*, 140, 1476–1495. [872,874,877]

Frühwirth-Schnatter, S. (1994), "Data Augmentation and Dynamic Linear Models," *Journal of Time Series Analysis*, 15, 183–202. [866]

Furrer, R. and Bengtsson, T. (2007), "Estimation of High-Dimensional Prior and Posterior Covariance Matrices in Kalman Filter Variants," *Journal of Multivariate Analysis*, 98, 227–255. [871]

Furrer, R., Genton, M. G., and Nychka, D. (2006), "Covariance Tapering for Interpolation of Large Spatial Datasets," *Journal of Computational and Graphical Statistics*, 15, 502–523. [871]

Gamerman, D. (1998), "Markov Chain Monte Carlo for Dynamic Generalized Linear Models," *Biometrika*, 85, 215–227. [866,867]

Gamerman, D., and Migon, H. S. (1993), "Dynamic Hierarchical Models," *Journal of the Royal Statistical Society*, Series B, 55, 629–642. [867]

Gelfand, A. E., and Smith, A. (1990), "Sampling-based Approaches to Calculating Marginal Densities," *Journal of the American Statistical Association*, 85, 398–409. [875]

Geman, S., and Geman, D. (1984), "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6, 721–741. [875]

Gneiting, T., and Katzfuss, M. (2014), "Probabilistic Forecasting," *Annual Review of Statistics and Its Application*, 1, 125–151. [874,879]

Goldstein, M., and Wooff, D. (2007). *Bayes Linear Statistics, Theory & Methods*, Chichester, UK: Wiley. [878]

Gordon, N., Salmond, D., and Smith, A. (1993), "Novel Approach to Nonlinear/non-Gaussian Bayesian State Estimation," *Radar and Signal Processing, IEE Proceedings F*, 140, 107–113. [866,869]

Gu, Y., and Oliver, D. S. (2007), "An Iterative Ensemble Kalman Filter for Multiphase Fluid Flow Data Assimilation," *SPE Journal*, 12, 438–446. [869]

Hartigan, J. (1969), "Linear Bayesian Methods," *Journal of the Royal Statistical Society*, Series B, 31, 446–454. [878]

Hastie, T., Tibshirani, R., and Friedman, J. (2009), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.), New York: Springer. [874]

Hinton, G. E. (2002). "Training Products of Experts by Minimizing Contrastive Divergence," *Neural Computation*, 14, 1771–1800. [879]

Houtekamer, P. L., Deng, X., Mitchell, H. L., Baek, S.-J., and Gagnon, N. (2014), "Higher Resolution in an Operational Ensemble Kalman Filter," *Monthly Weather Review*, 142, 1143–1162. [867,871,878]

Houtekamer, P. L., and Mitchell, H. L. (2005), "Ensemble Kalman Filtering. *Quarterly Journal of the Royal Meteorological Society*, 131, 3269–3289. [871,878]

Houtekamer, P. L., and Zhang, F. (2016), "Review of the Ensemble Kalman Filter For Atmospheric Data Assimilation," *Monthly Weather Review*, 144, 4489–4532. [866]

Hunt, B. R., Kostelich, E. J., and Szunyogh, I. (2007), "Efficient Data Assimilation for Spatiotemporal Chaos: A Local Ensemble Transform Kalman Filter," *Physica D: Nonlinear Phenomena*, 230, 112–126. [871]

Jiang, W., and Tanner, M. A. (2008), "Gibbs Posterior for Variable Selection in High-Dimensional Classification and Data Mining," *Annals of Statistics*, 36, 2207–2231. [874]

Johansen, A. M., Whiteley, N., and Doucet, A. (2012), "Exact Approximation of Rao-Blackwellised Particle Filters," in *IFAC Proceedings Volumes* (vol. 45), pp. 488–493. [869,872,876,880]

Kalman, R. (1960), "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, 82, 35–45. [866,869]

Katzfuss, M., and Bhattacharya, A. (2015), "Comment on Article by Dawid and Musio," *Bayesian Analysis*, 10, 501–504. [874]

Katzfuss, M., Stroud, J. R., and Wikle, C. K. (2016), "Understanding the Ensemble Kalman Filter," *The American Statistician*, 70, 350–357. [866, 870]

Khare, S., Anderson, J., Hoar, T., and Nychka, D. (2008), "An Investigation Into the Application of an Ensemble Kalman Smoother to High-Dimensional Geophysical Systems," *Tellus*, 60A, 97–112. [871]

Kitagawa, G. (1996), "Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models," *Journal of Computational and Graphical Statistics*, 5, 1–25. [869,878]

Law, K. J. H., Tembine, H., and Tempone, R. (2016), "Deterministic Mean-Field Ensemble Kalman Filtering," *SIAM Journal on Scientific Computing*, 38, A1251–A1279. [878]

Le Gland, F., Monbet, V., and Tran, V.-D. (2011), "Large Sample Asymptotics for the Ensemble Kalman Filter," in *The Oxford Handbook of Nonlinear Filtering*, eds. Crisan, D., and Rozovskii, B., 598–634, Oxford, UK: Oxford University Press. [878]

Lei, J., and Bickel, P. (2011), "A Moment Matching Ensemble Filter for Nonlinear Non-Gaussian Data Assimilation," *Monthly Weather Review*, 139, 3964–3973. [867]

Lei, J., Bickel, P. J., and Snyder, C. (2010), "Comparison of Ensemble Kalman Filters Under Non-Gaussianity," *Monthly Weather Review*, 138, 1293–1306. [871]

Li, C., Jiang, W., and Tanner, M. A. (2013), "General Oracle Inequalities for Gibbs Posterior With Application to Ranking," *Journal of Machine Learning Research: Workshop and Conference Proceedings*, vol. 30, pp. 512–521. [874]

Lindgren, F., Rue, H., and Lindström, J. (2011), "An Explicit Link Between Gaussian Fields and Gaussian Markov Random Fields: The Stochastic Partial Differential Equation Approach," *Journal of the Royal Statistical Society*, Series B, 73, 423–498. [869]

Liu, J., and West, M. (2001), "Combined Parameter and State Estimation in Simulation-Based Filtering," in *Sequential Monte Carlo Methods in Practice*, eds. A. Doucet, J. de Freitas, and N. Gordon, New York: Springer. [877]

Liu, J. S., and Chen, R. (1998), "Sequential Monte Carlo Methods for Dynamic Systems," *Journal of the American Statistical Association*, 93, 1032–1042. [866]

Lorenz, E. (1996), "Predictability: A Problem Partially Solved," in *Proceedings Seminar on Predictability*, pp. 1–18. ECMWF: Reading Berkshire, UK. [868,881]

Mandel, J., Cobb, L., and Beezley, J. D. (2011), "On the Convergence of the Ensemble Kalman Filter," *Applications of Mathematics*, 56, 533–541. [883]

Marin, J., Pudlo, P., Robert, C., and Ryder, R. (2012), "Approximate Bayesian Computational Methods," *Statistics and Computing*, 22, 1167–1180. [869]

Mitchell, H. L., and Houtekamer, P. L. (2000), "An Adaptive Ensemble Kalman Filter," *Monthly Weather Review*, 128, 416–433. [874]

Myrseth, I., and Omre, H. (2010), "Hierarchical Ensemble Kalman Filter," *SPE Journal*, 15, 569–580. [876]

Nott, D. J., Fan, Y., Marshall, L., and Sisson, S. A. (2014), "Approximate Bayesian Computation and Bayes' Linear Analysis: Toward High-Dimensional ABC," *Journal of Computational and Graphical Statistics*, 23, 65–86. [878]

Nott, D. J., Marshall, L., and Ngoc, T. M. (2012), "The Ensemble Kalman Filter is an ABC Algorithm," *Statistics and Computing*, 22, 1273–1276. [878]

Ott, E., Hunt, B. R., Szunyogh, I., Zimin, A. V., Kostelich, E. J., Corazza, M., Kalnay, E., Patil, D. J., and Yorke, J. A. (2004), "A Local Ensemble Kalman Filter for Atmospheric Data Assimilation," *Tellus*, 56A, 415–428. [871]

Pfanzagl, J. (1969), "On the Measurability and Consistency of Minimum Contrast Estimates," *Metrika*, 14, 249–272. [874]

Polson, N. G., Stroud, J. R., and Müller, P. (2008), "Practical Filtering With Sequential Parameter Learning," *Journal of the Royal Statistical Society*, Series B, 70, 413–428. [869]

Pourahmadi, M. (2013). *High-Dimensional Covariance Estimation*, Hoboken, NJ: Wiley. [871]

Raftery, A. E., Newton, M. A., Satagopan, J. M., and Krivitsky, P. N. (2007). "Estimating the Integrated Likelihood via Posterior Simulation Using the Harmonic Mean Identity," in *Bayesian Statistics*, eds. J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith, and M. West (Vol. 8, pp. 1–45), Oxford, UK: Oxford University Press. [874]

Rao, V., Sandu, A., Ng, M., and Nino-Ruiz, E. (2017), "Robust Data Assimilation Using L_1 and Huber Norms," *SIAM Journal on Scientific Computing*, 39, B548–B570. [875]

Robert, C. and Casella, G. (2004). *Monte Carlo Statistical Methods*, New York: Springer. [883]

Roh, S., Genton, M. G., Jun, M., Szunyogh, I., and Hoteit, I. (2013), "Observation Quality Control With a Robust Ensemble Kalman Filter," *Monthly Weather Review*, 141, 4414–4428. [879,880]

Rue, H., Martino, S., and Chopin, N. (2009), "Approximate Bayesian Inference for Latent Gaussian Models by Using Integrated Nested Laplace Approximations," *Journal of the Royal Statistical Society*, Series B, 71, 319–392. [869,874]

Sætrom, J., and Omre, H. (2011), "Ensemble Kalman Filtering With Shrinkage Regression Techniques," *Computational Geosciences*, 15, 271–292. [871]

——— (2013), "Uncertainty Quantification in the Ensemble Kalman Filter," *Scandinavian Journal of Statistics*, 40(4):868–885. [871]

Sansó, B., and Guenni, L. (1999), "Venezuelan Rainfall Data Analysed by Using a Bayesian Space-Time Model," *Applied Statistics*, 48, 345–362. [867,868]

Schweppe, F. (1965), "Evaluation of Likelihood Functions for Gaussian Signals," *IEEE Transactions on Information Theory*, 11, 61–70. [869]

Shaby, B. A. (2014), "The Open-faced Sandwich Adjustment for MCMC Using Estimating Functions," *Journal of Computational and Graphical Statistics*, 23, 853–876. [874]

Shephard, N., and Pitt, M. (1997), "Likelihood Analysis of Non-Gaussian Measurement Time Series," *Biometrika*, 84, 653–667. [866,867]

Shumway, R. H., and Stoffer, D. S. (2006). *Time Series Analysis and its Applications With R Examples* (2nd ed.). Springer, New York. [866,867]

Sigrist, F., Künsch, H. R., and Stahel, W. A. (2012), "A Dynamic Nonstationary Spatio-Temporal Model for Short Term Prediction of Precipitation," *Annals of Applied Statistics*, 6, 1452–1477. [868]

Sisson, S. A., Fan, Y., and Tanaka, M. M. (2007), "Sequential Monte Carlo Without Likelihoods," *Proceedings of the National Academy of Sciences*, 104, 1760–1765. [869]

Snyder, C., Bengtsson, T., Bickel, P., and Anderson, J. L. (2008), "Obstacles to High-Dimensional Particle Filtering," *Monthly Weather Review*, 136, 4629–4640. [866,870,876]

Stroud, J. R., and Bengtsson, T. (2007), "Sequential State and Variance Estimation Within the Ensemble Kalman Filter," *Monthly Weather Review*, 135, 3194–3208. [867]

Stroud, J. R., Katzfuss, M., and Wikle, C. K. (2018), "A Bayesian Adaptive Ensemble Kalman Filter for Sequential State and Parameter Estimation," *Monthly Weather Review*, 146, 373–386. [874]

Stroud, J. R., Stein, M. L., Lesht, B. M., Schwab, D. J., and Beletsky, D. (2010), "An Ensemble Kalman Filter and Smoother for Satellite Data Assimilation," *Journal of the American Statistical Association*, 105, 978–990. [871,874,878]

Szunyogh, I., Kostelich, E. J., Gyarmati, G., Kalnay, E., Hunt, B. R., Ott, E., Satterfield, E., and Yorke, J. A. (2008), "A Local Ensemble Transform Kalman Filter Data Assimilation System for the NCEP Global Model," *Tellus*, 60A, 113–130. [867]

Tierney, L., and Kadane, J. B. (1986), "Accurate Approximations for Posterior Moments and Marginal Densities," *Journal of the American Statistical Association*, 81, 82–86. [874]

Tippett, M. K., Anderson, J. L., Bishop, C. H., Hamill, T. M., and Whitaker, J. S. (2003), "Ensemble Square-root Filters," *Monthly Weather Review*, 131, 1485–1490. [878]

Tsyrulnikov, M., and Rakitko, A. (2017), "Hierarchical Bayes Ensemble Kalman Filtering," *Physica D*, 338, 1–16. [876]

Ueno, G., and Nakamura, N. (2014), "Iterative Algorithm for Maximum-Likelihood Estimation of the Observation-Error Covariance Matrix for Ensemble-Based Filters," *Quarterly Journal of the Royal Meteorological Society*, 140, 295–315. [872]

——— (2016), "Bayesian Estimation of Observation Error Covariance Matrix in Ensemble-Based Filters," *Quarterly Journal of the Royal Meteorological Society*, 142, 2055–2080. [872]

Urbanek, S., and Rubner, Y. (2012). *emdist: Earth Mover's Distance*. R package version 0.3-1. [881]

van Leeuwen, P. J., and Evensen, G. (1996), "Data Assimilation and Inverse Methods in Terms of a Probabilistic Formulation," *Monthly Weather Review*, 124, 2898–2913. [871]

Wald, A. (1949), "Note on the Consistency of the Maximum Likelihood Estimate," *The Annals of Mathematical Statistics*, 20, 595–601. [874]

Wang, X., and Bishop, C. H. (2003), "A comparison of Breeding and Ensemble Transform Kalman Filter Ensemble Forecast Schemes," *Journal of the Atmospheric Sciences*, 60, 1140–1158. [868]

West, M. (1987), "On Scale Mixtures of Normality," *Biometrika*, 74, 694–697. [867]

West, M. and Harrison, P. (1997), *Bayesian Forecasting and Dynamic Models* (2nd ed.). New York: Springer . [866]

Whitaker, J. S., Hamill, T. M., Wei, X., Song, Y., and Toth, Z. (2008), "Ensemble Data Assimilation With the NCEP Global Forecast System," *Monthly Weather Review*, 136, 463–482. [867]

Wikle, C. K. (2002), "A Kernel-based Spectral Model for Non-Gaussian Spatial Processes," *Statistical Modelling: An International Journal*, 2, 299–314. [868,880]

Wikle, C. K., and Berliner, L. M. (2007), "A Bayesian Tutorial for Data Assimilation," *Physica D: Nonlinear Phenomena*, 230, 1–16. [866]

Wood, S. N. (2010), "Statistical Inference for Noisy Nonlinear Ecological Dynamic Systems," *Nature*, 466, 1102–1104. [869]