Model error covariance estimation in particle and ensemble Kalman filters using an online expectation-maximization algorithm

Tadeo Javier Cocucci, Manuel Pulido, Magdalena Lucini, Pierre Tandeo

MODEL ERROR COVARIANCE ESTIMATION IN PARTICLE AND ENSEMBLE KALMAN FILTERS USING AN ONLINE EXPECTATION-MAXIMIZATION ALGORITHM

Tadeo Javier Cocucci
FaCENA, Universidad Nacional del Nordeste
Corrientes, Corrientes, 3400, Argentina
tadeojcocucci@gmail.com

Manuel Pulido
FaCENA, Universidad Nacional del Nordeste
Corrientes, Corrientes, 3400, Argentina
CONICET, Corrientes, Argentina

Magdalena Lucini
FaCENA, Universidad Nacional del Nordeste
Corrientes, Corrientes, 3400, Argentina
CONICET, Corrientes, Argentina

Pierre Tandeo
IMT Atlantique,
Lab-STICC, UMR CNRS 6285
F-29238, France

March 5, 2020

ABSTRACT

The performance of ensemble-based data assimilation techniques that estimate the state of a dynamical system from partial observations depends crucially on the prescribed uncertainty of the model dynamics and of the observations. These are not usually known and have to be inferred. Many approaches have been proposed to tackle this problem, including fully Bayesian, likelihood maximization and innovation-based techniques. This work focuses on maximization of the likelihood function via the expectation-maximization (EM) algorithm to infer the model error covariance combined with ensemble Kalman filters and particle filters to estimate the state. The classical application of the EM algorithm in a data assimilation context involves filtering and smoothing a fixed batch of observations in order to complete a single iteration. This is an inconvenience when using sequential filtering in high-dimensional applications. Motivated by this, an adaptation of the algorithm that can process observations and update the parameters on the fly, with some underlying simplifications, is presented. The proposed technique was evaluated and achieved good performance in experiments with the Lorenz-63 and the 40-variable Lorenz-96 dynamical systems designed to represent some common scenarios in data assimilation such as non-linearity, chaoticity and model misspecification.

***K*eywords** Uncertainty quantification · Parameter estimation · Model error · Expectation-Maximization

1 Introduction

Data assimilation techniques including variational data assimilation, ensemble Kalman filters (EnKF) and particle filters estimate the state of a time-varying process combining constraints on the dynamics of the system and some observations of the process. Both the observation process and the dynamical model of the state are prone to errors. Model error encompasses our incomplete and possibly truncated knowledge of the dynamics including discretization errors. On the other hand, observational error includes the inaccuracy of the measurement instrument and the possible misrepresentation of the observational operator that maps the state to the observations, also called representativity error. Data assimilation relies on the specification of these errors to assess the quality of the information provided by the forecast, given by the model dynamics, and by the observations in order to provide an estimate of the state that considers both sources of information. However, model and observational errors are not usually available so they have to be estimated. A variety of methods has been proposed to do this, including Bayesian approaches, innovation-based

methods and likelihood maximization techniques, among others. A recent and detailed review of these methods can be found in Tandeo et al. (2018).

In particle or ensemble-based filtering techniques, the model error covariance is related to the spread of the particles –ensemble members– representing the forecasted state. Using inadequate estimations of it may lead to a misrepresentation of the forecast uncertainty which can produce filter underperformance (when the spread is overdispersive), or in some cases, its divergence (when the spread is underdispersive). Even if the true model error covariance is used, it is well known (as shown in Houtekamer and Mitchell, 1998) that due to subsampling the ensemble has a tendency to collapse and possibly lead to filter degeneracy. This effect is often alleviated through a multiplicative or additive covariance inflation factor (Anderson and Anderson, 1999; Miyoshi, 2011). Since the effective model error covariance impacts on the ensemble spread, the problem of estimating these uncertainties becomes important when using the EnKF.

In the case of particle filters, model error covariance impacts directly on the uncertainty of each (independent) particle. Recently, some ideas have emerged to improve the sampling of particle filters in high-dimensional spaces guiding the particles toward regions of high likelihood. These approaches include the implicit particle filter (Chorin and Tu, 2009; Atkins et al., 2013), the implicit equal-weight particle filter (Zhu et al., 2016), the particle flow filter based on tempering (Daum and Huang, 2009) and the variational mapping particle filter (Pulido and van Leeuwen, 2018). An implicit assumption in all these filters is the knowledge of the model error covariance. In other words, the knowledge of the uncertainty associated to each particle. Therefore, these filters require in practice a method to infer the model error covariance in complex systems.

There are several applications in which the estimation of the state of the system is a big challenge because of the high-dimensionality (Snyder et al., 2008). Estimation of model uncertainty is even more challenging in terms of dimensionality. In several scenarios model error is represented through the covariance matrix $\mathbf{Q}$ of a multivariate Gaussian distribution, so if the dimension of the state is N_x then $\mathbf{Q}$ is a $N_x \times N_x$ matrix. Hence, the estimation of the full covariance matrix $\mathbf{Q}$ becomes infeasible. To make this estimation problem achievable within a data assimilation context, a commonly used approach is to parametrize the matrix or assume it has a certain shape, such as band-diagonal. These assumptions may be motivated physically, e.g. physical space localization of the covariance (Hamill et al., 2001; Ueno et al., 2010) or correlation functions in the sphere (Gaspari and Cohn, 1999).

In a similar fashion, the observational error covariance is often represented by a $N_y \times N_y$ covariance matrix $\mathbf{R}$ of a multivariate Gaussian distribution. Although N_y is usually much smaller than N_x parameterizations and fixed structure assumptions can be used to simplify the problem of observational errror covariance estimation and to diminish computational costs. For example, in Stroud et al. (2018), a Matérn covariance model is used to represent spatially correlated observation errors.

Some of the methods developed to estimate these error covariances focus on finding the parameters that maximize the likelihood of the observations which is usually computationally expensive to evaluate or approximate. In a seminal work, Shumway and Stoffer (1982) proposed to use the expectation-maximization (henceforth EM) algorithm (Dempster et al., 1977) in combination with the linear Kalman filter, to produce a sequence of estimates that converges to the maximum likelihood estimator. More recent applications of EM in EnKFs (Tandeo et al., 2015; Dreano et al., 2017; Pulido et al., 2018) use a fixed batch of observations that are assimilated with a forward EnKF and a backwards ensemble Kalman Smoother in order to complete one iteration of the EM algorithm. This batch EM approach was extended by Lucini et al. (2019) for particle filters based on the innovation likelihood to avoid the need of a particle smoother in the likelihood function. The batch EM has a high-computational cost in high-dimensional spaces. Furthermore, the requirement to store and to reuse every observation within the batch is inconvenient in the context of sequential filtering.

On the other hand, a variety of sequential or *online* methods to estimate error covariances has been developed. These produce updates ofthe parameter estimations sequentially, at each assimilation cycle/filtering step, so that there is no need to store all previous observations or to smooth backwards through the whole batch of observations. An online version of the EM algorithm has been proposed for sequential importance resampling (SIR) particle filters in Cappé (2009), whereas Andrieu and Doucet (2003) also provided an online version of EM for state-space models based on a split-data likelihood function which combines pseudo-likelihood values computed with different mini-batches of data. Based on these ideas, we propose here an efficient EM-based online algorithm which can be coupled with both particle filters and EnKFs. Its performance is assessed in experiments of interest in geophysical applications.

The rest of this work is outlined as follows. In Section 2, we derive the proposed online method starting from the batch EM algorithm. Section 3 describes the experiments for which we evaluated the method's performance and examine the results. In Section 4, we draw the conclusions and give perspectives of possible future work. The appendices

include some specification of the dynamical systems (Appendix A) and the filtering techniques used in the experiments (Appendix B).

2 Theoretical derivation

We consider a hidden Markov model consisting in a dynamical model $\mathcal{M}$ for the hidden state variables $\mathbf{x}_k$ and observations of this state $\mathbf{y}_k$ at time k. The latter are represented through an observational function $\mathcal{H}$ both with additive noise processes. Namely,

$$\mathbf{x}_k = \mathcal{M}(\mathbf{x}_{k-1}) + \boldsymbol{\eta}_k, \tag{1}$$

$$\mathbf{y}_k = \mathcal{H}(\mathbf{x}_k) + \boldsymbol{\nu}_k, \tag{2}$$

where the model and observational noise are realizations from $\boldsymbol{\eta}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$ and $\boldsymbol{\nu}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ respectively.

Data assimilation techniques involve estimating the density of the hidden state given a set of observations, $p(\mathbf{x}_k|\mathbf{y}_{1:k})$, from Equations 1 and 2 but rely on knowing the parameters of the state space model including $\mathcal{M}$, $\mathcal{H}$, $\mathbf{R}_k$, $\mathbf{Q}_k$ and the initial state density $p(\mathbf{x}_0)$. The methodology to be described may estimate all of these parameters, which we denote in general by $\boldsymbol{\theta}$. However, in this work, the dynamical model $\mathcal{M}$, observational operator $\mathcal{H}$ and $p(\mathbf{x}_0)$ are assumed to be known, and we focus on the uncertainty quantification of the hidden Markov model –the estimation of the error covariances $\mathbf{R}_k$ and $\mathbf{Q}_k$ –, particularly the latter. The role of the initial state density, $p(\mathbf{x}_0)$, is not expected to be important in sequential assimilation apart from some initial spinup time.

2.1 Batch EM algorithm

The aim of the traditional batch EM algorithm applied to a hidden Markov model is to iteratively find the statistical parameters $\boldsymbol{\theta}$ that maximize the complete likelihood function $p(\mathbf{y}_{1:K}; \boldsymbol{\theta})$, given a batch of observations distributed in a time interval, $\mathbf{y}_{1:K} \triangleq \{\mathbf{y}_1, \ldots, \mathbf{y}_K\}$. Since the realizations of the state $\mathbf{x}_{0:K}$ are unknown, we write the likelihood function in a marginalized form,

$$p(\mathbf{y}_{1:K}; \boldsymbol{\theta}) = \int p(\mathbf{x}_{0:K}, \mathbf{y}_{1:K}; \boldsymbol{\theta}) \mathrm{d}\mathbf{x}_{0:K}, \tag{3}$$

The logarithm of this likelihood function may be reexpressed in terms of an arbitrary density $q(\mathbf{x}_{0:K})$ (Neal and Hinton, 1998) with the sole constrain that its support contains the support of the conditional density $p(\mathbf{x}_{0:K}|\mathbf{y}_{1:K})$. Using the conditional density definition and that $\int q(x_{0:K}) dx_{0:K} = 1$, the loglikelihood function can be written as,

$$\begin{aligned} \log p(\mathbf{y}_{1:K}; \boldsymbol{\theta}) &= \mathcal{D}_{KL}(q(\mathbf{x}_{0:K}) \| p(\mathbf{x}_{0:K}|\mathbf{y}_{1:K}; \boldsymbol{\theta})) + \int q(\mathbf{x}_{0:K}) \log \frac{p_\theta(\mathbf{x}_{0:K}, \mathbf{y}_{1:K}; \boldsymbol{\theta})}{q(\mathbf{x}_{0:K})} \mathrm{d}\mathbf{x}_{0:K}, \\ &\triangleq \mathcal{D}_{KL}(q\|p) + \mathcal{L}(q, \boldsymbol{\theta}), \end{aligned} \tag{4}$$

where the Kullback-Leibler divergence ($\mathcal{D}_{KL}(q|p) \triangleq \int q(x) \log[q(x)/p(x)] \mathrm{d}x$) is a non-negative function so that $\mathcal{L}(q, \boldsymbol{\theta})$ is a lower bound of the loglikelihood, and is thus referred to as the evidence lower bound or ELBO (Goodfellow et al., 2016). The upper bound of the ELBO is attained at $q(\mathbf{x}_{0:K}) = p(\mathbf{x}_{0:K}|\mathbf{y}_{1:K}; \boldsymbol{\theta})$ for which $\mathcal{D}_{KL}(q|p) = 0$. In this upper limit, the ELBO equals the loglikelihood function.

Given an initial guess $\boldsymbol{\theta}_0$, the i-th iteration of the EM algorithm encompasses

- *Expectation step.* Maximize the ELBO as a function of q given a set of parameters $\boldsymbol{\theta}_{i-1}$, i.e. $q_{i-1} = \arg\max_q \mathcal{L}(q, \boldsymbol{\theta}_{i-1})$. The resulting optimal density is $q_{i-1}(\mathbf{x}_{0:K}) = p(\mathbf{x}_{0:K}|\mathbf{y}_{1:K}; \boldsymbol{\theta}_{i-1})$.
- *Maximization step.* Maximize the ELBO w.r.t. $\boldsymbol{\theta}$ given a fixed q_{i-1}, i.e. $\boldsymbol{\theta}_i = \arg\max_{\boldsymbol{\theta}} \mathcal{L}(q_{i-1}, \boldsymbol{\theta})$.

Since the methodology guarantees that $\mathcal{L}(q, \boldsymbol{\theta})$ increases at each EM iteration (or remains unchanged if a maximum has been reached), while $\mathcal{D}_{KL}(q_i(\mathbf{x}_{0:K}) \| p(\mathbf{x}_{0:K}|\mathbf{y}_{1:K}; \boldsymbol{\theta}))$ decreases (or remains unchanged), the algorithm converges to a set of parameters that (locally) maximizes the loglikelihood (Wu and Jeff, 1983). In other words, the approximation function $q(\mathbf{x}_{0:K})$ during the convergence gets closer to the posterior density given the optimal parameters as the ELBO increases and therefore the Kullback-Leibler divergence between these two probability density functions decreases.

A practical aspect of the algorithm is that the two steps do not require to reach the maximum of the ELBO to guarantee convergence. As long as we find a $\boldsymbol{\theta}_i$ that increases the ELBO in the maximization step (this family of methods are called generalized EM algorithms) and a function q_{i-1} that increases the ELBO in the expectation step (called as incremental EM algorithms), the convergence is guaranteed.

2.2 The batch EM algorithm in a hidden Markov model

A central property of the hidden Markov model is that the joint density can be expressed as a product. Using the Markov property of the state, and that observations only depend on the current state, the joint probability density in a time interval $0{:}K$ (e.g. Pulido et al., 2018) results in

$$\begin{aligned} p(\mathbf{x}_{0:K}, \mathbf{y}_{1:K}; \boldsymbol{\theta}) &= p(\mathbf{x}_0; \boldsymbol{\theta}) \prod_{k=1}^{K} p(\mathbf{x}_k|\mathbf{x}_{k-1}; \boldsymbol{\theta}) \prod_{k=1}^{K} p(\mathbf{y}_k|\mathbf{x}_k; \boldsymbol{\theta}) \\ &= p(\mathbf{x}_0; \boldsymbol{\theta}) \prod_{k=1}^{K} p(\mathbf{x}_k, \mathbf{y}_k|\mathbf{x}_{k-1}; \boldsymbol{\theta}). \end{aligned} \tag{5}$$

Using the product form of the joint density, in Equation 5, the resulting ELBO function for hidden Markov models is

$$\begin{aligned} \mathcal{L}(p(\mathbf{x}_{0:K}|\mathbf{y}_{1:K}; \boldsymbol{\theta}_{i-1}), \boldsymbol{\theta}) &= \sum_{k=1}^{K} \int p(\mathbf{x}_{k-1:k}|\mathbf{y}_{1:K}; \boldsymbol{\theta}_{i-1}) \log p(\mathbf{x}_k, \mathbf{y}_k|\mathbf{x}_{k-1}; \boldsymbol{\theta}) \mathrm{d}\mathbf{x}_{k-1:k} + C, \\ &\triangleq \sum_{k=1}^{K} \mathcal{E}\Big[\log p(\mathbf{x}_k, \mathbf{y}_k|\mathbf{x}_{k-1}) \Big| \mathbf{y}_{1:K}; \boldsymbol{\theta}_{i-1} \Big] + C, \end{aligned} \tag{6}$$

where all the constant terms w.r.t. $\boldsymbol{\theta}$ are included in C and dropped from $\mathcal{L}$ in what follows. Furthermore, note that $\mathcal{E}\Big[f(\mathbf{x}_{k-1}, \mathbf{x}_k) \Big| \mathbf{y}_{1:K}; \boldsymbol{\theta} \Big] = \int f(\mathbf{x}_{k-1}, \mathbf{x}_k) p(\mathbf{x}_{0:K}|\mathbf{y}_{1:K}; \boldsymbol{\theta}) \mathrm{d}\mathbf{x}_{k-1:k} = \int f(\mathbf{x}_{k-1}, \mathbf{x}_k) p(\mathbf{x}_{k-1}, \mathbf{x}_k|\mathbf{y}_{1:K}; \boldsymbol{\theta}) \mathrm{d}\mathbf{x}_{k-1:k}$.

In the general case, the parameters that maximize the ELBO in a hidden Markov model in the maximization step, need to be determined numerically. However, if $p(\mathbf{x}_k, \mathbf{y}_k|\mathbf{x}_{k-1})$ belongs to an exponential family (condition satisfied when both observational and model errors belong to the exponential family), it is possible to derive an analytical expression for the parameters that maximize the ELBO (Equation 6). This is a rather general parametric density family assumption which includes the usual Gaussian model and observational errors, to be specified below. The joint density of state and observation given the previous state required in Equation 5 is in that case expressed as

$$p(\mathbf{x}_k, \mathbf{y}_k|\mathbf{x}_{k-1}; \boldsymbol{\theta}) = h(\mathbf{x}_k, \mathbf{y}_k) \exp\left[\psi(\boldsymbol{\theta}) \cdot s(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{y}_k) - A(\boldsymbol{\theta}) \right], \tag{7}$$

where $s(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{y}_k)$ is the natural sufficient statistic, $\psi(\boldsymbol{\theta})$ is called the natural parameter and h and A are functions.

The gradient of the ELBO w.r.t. $\boldsymbol{\theta}$ by introducing Equation 7 into 6 is

$$\nabla_\theta \mathcal{L}(p(\mathbf{x}_{0:K}|\mathbf{y}_{1:K}; \boldsymbol{\theta}_{i-1}), \boldsymbol{\theta}) = \nabla_\theta \psi(\boldsymbol{\theta}) \cdot \sum_{k=1}^{K} \mathcal{E}\Big[s(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{y}_k) \Big| \mathbf{y}_{1:K}; \boldsymbol{\theta}_{i-1} \Big] - K \nabla_\theta A(\boldsymbol{\theta}). \tag{8}$$

Assuming the expression in Equation 8 has one root, corresponding to the maximum of $\mathcal{L}$, the resulting E-step of the i-th iteration becomes

$$S_{i-1} = \frac{1}{K} \sum_{k=1}^{K} \mathcal{E}\Big[s(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{y}_k) \Big| \mathbf{y}_{1:K}; \boldsymbol{\theta}_{i-1} \Big], \tag{9}$$

while the M-step is

$$\boldsymbol{\theta}_i = \widehat{\boldsymbol{\theta}}\left(S_{i-1} \right), \tag{10}$$

where we define $\widehat{\boldsymbol{\theta}}(S_{i-1})$ as the solution for $\boldsymbol{\theta}$ of the equation $\nabla_\theta \psi(\boldsymbol{\theta}) \cdot S_{i-1} - K \nabla_\theta A(\boldsymbol{\theta}) = 0$.

In the case that both the observational and the transition densities are Gaussian

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{x}_{k-1}) &= \mathcal{N}(\mathcal{M}(\mathbf{x}_{k-1}), \mathbf{Q}), \\ p(\mathbf{y}_k|\mathbf{x}_k) &= \mathcal{N}(\mathcal{H}(\mathbf{x}_k), \mathbf{R}), \end{aligned}$$

and the parameters to estimate are their respective covariances $\boldsymbol{\theta} = \{\mathbf{Q}, \mathbf{R}\}$ which are assumed constant in time, the quantity S_{i-1} can be expressed as a tuple (S^Q_{i-1}, S^R_{i-1}). Using Equations 8 and 9 we get the corresponding expressions,

$$S^Q_{i-1} = \frac{1}{K}\sum_{k=1}^{K} \mathcal{E}\Big[(\mathbf{x}_k - \mathcal{M}(\mathbf{x}_{k-1}))(\mathbf{x}_k - \mathcal{M}(\mathbf{x}_{k-1}))^\top \Big| \mathbf{y}_{1:K}; \boldsymbol{\theta}_{i-1}\Big], \tag{11}$$

$$S^R_{i-1} = \frac{1}{K}\sum_{k=1}^{K} \mathcal{E}\Big[(\mathbf{y}_k - \mathcal{H}(\mathbf{x}_k))(\mathbf{y}_k - \mathcal{H}(\mathbf{x}_k))^\top \Big| \mathbf{y}_{1:K}; \boldsymbol{\theta}_{i-1}\Big]. \tag{12}$$

In this particlar case, because of the Gaussian assumption, the maximization with respect to the parameters in Equation 10 results in $(\mathbf{Q}_i, \mathbf{R}_i) = \widehat{\boldsymbol{\theta}}(S_{i-1}) = S_{i-1} = (S^Q_{i-1}, S^R_{i-1})$, which amounts to say that $\widehat{\boldsymbol{\theta}}$ is the identity function.

The update formulas in Equations 11 and 12 are the main calculations used in the implementations of batch EM such as Tandeo et al. (2015); Dreano et al. (2017); Pulido et al. (2018). These expressions involve computing the smoothed densities $p(\mathbf{x}_k|\mathbf{y}_{1:K})$ for every k at each EM iteration through Kalman smoothing techniques in the Gaussian linear hidden Markov model. In other words, this requires to process the whole batch of observations in each iteration of the EM algorithm. Furthermore, if new observations are added, the whole process needs to be redone. The batch EM is a robust way to estimate the global structure of $\mathbf{Q}$ and $\mathbf{R}$. However, since it can be very costly and unsuitable in the context of observations gathered sequentially in time, we aim to develop an "online" approach for high-dimensional systems based on the EM algorithm that enables to update the parameters *at each* assimilation cycle. The goal of doing this is to avoid to store all of the observations and to reduce substantially the computational cost, since each observation would be processed only once. Besides, online techniques may track time-varying error covariances which is important for geophysical applications.

2.3 The online EM algorithm

The major drawback of batch EM is that for each new observation, and a given value of the parameters $\boldsymbol{\theta}$, the ELBO has to be recomputed using the whole sequence of observations from 1 to K. Our goal is to update the parameter with each new observation. Thus, in what follows the EM iteration index is dropped because, in this sequential context, it coincides with K, the index of the last observation being processed. With this in mind, we will derive here an approximated recursive form for the required quantity in Equation 9. We begin by writing it as:

$$S_K = \frac{1}{K}\left(\sum_{k=1}^{K-1}\int s(\mathbf{x}_{k-1}, \mathbf{x}_k, \mathbf{y}_k) p(\mathbf{x}_{k-1}, \mathbf{x}_k|\mathbf{y}_{1:K}; \boldsymbol{\theta}_K) d\mathbf{x}_{k-1:k} + \right.$$
$$\left. \int s(\mathbf{x}_{K-1}, \mathbf{x}_K, \mathbf{y}_K) p(\mathbf{x}_{K-1}, \mathbf{x}_K|\mathbf{y}_{1:K}; \boldsymbol{\theta}_K) d\mathbf{x}_{K-1:K}\right). \tag{13}$$

We can recognize that the first $K-1$ terms in Equation 13 correspond to $(K-1)S_{K-1}$ but incorporating information of the newly available observation $\mathbf{y}_K$ so that the posterior density corresponds to smoothing. We make the approximation that $\mathbf{y}_K$ does not significantly influence the previous state estimates but only provides information to the last term, which corresponds to the sufficient statistics at K. This results in:

$$\tilde{S}_K = (1-\gamma_K)\tilde{S}_{K-1} + \gamma_K \int s(\mathbf{x}_{K-1}, \mathbf{x}_K, \mathbf{y}_K) p(\mathbf{x}_{K-1}, \mathbf{x}_K|\mathbf{y}_{1:K}; \boldsymbol{\theta}_K) d\mathbf{x}_{K-1:K} \tag{14}$$

$$= (1-\gamma_K)\,\tilde{S}_{K-1} + \gamma_K \mathcal{E}\left[s(\mathbf{x}_{K-1}, \mathbf{x}_K, \mathbf{y}_K)|\mathbf{y}_{1:K}; \boldsymbol{\theta}_K\right], \tag{15}$$

where we introduced $\gamma_k \in (0, 1)$ as a step-size (instead of $1/K$), based on stochastic approximation techniques (LeGland and Mevel, 1997). This parameter controls the memory of the statistics, determining the importance of the old statistics relative to the contribution of the current observation. The initialization parameter $\tilde{S}_0$ has to be provided as a first guess. In the Gaussian case, it coincides with the first guess for the parameter $\boldsymbol{\theta}_0$.

There is some resemblance with the technique proposed by Cappé (2009) where, in order to keep track of the statistics, an auxiliary function needs to be kept updated. This auxiliary function relates to a recursive form of smoothing and can be used to propagate the information of the latest observation to the previous estimate of the statistic. In our algorithm, γ_k controls the influence of previous estimate on the current one, so under the proposed assumption, we bypass the need of using this auxiliary function. The auxiliary function in Cappé (2009) makes an exact EM update. On the other hand, the algorithm here proposed, is an approximation to account for just a single smoothing step. A

stronger assumption was taken in Lucini et al. (2019), they proposed an EM batch algorithm which only uses the filtering posterior density without accounting the effect of subsequent observations during the parameter estimation.

To compute the expectation in Equation 15, we propose two related approaches: importance sampling integration implemented in such a way that no smoothing is required (suitable for particle filters) and direct Monte Carlo integration using a sample of the required joint smoothing density $p(\mathbf{x}_{k-1:k}|\mathbf{y}_{1:k})$. These two methods are further explained in the following subsections.

In the batch EM, the parameters are fixed during the whole batch of observations and are updated in each EM iteration. On the other hand, the parameters are updated in this online EM version at each observation time when information on the current observations is available. On the other hand, in this online version of the EM algorithm the parameters are updated when a new observation is available, that is, at each observation time. Thus, while batch EM assumes that parameters are fixed, in the present scheme we assume parameters change slowly as in hierarchical Bayesian approaches (Scheffler et al., 2019).

2.3.1 Importance sampling

In order to compute the expectation in Equation 15, we write it in integral form:

$$\mathcal{E}\left[s(\mathbf{x}_{K-1}, \mathbf{x}_K, \mathbf{y}_K)|\mathbf{y}_{1:K}; \boldsymbol{\theta}_K\right] = \int s(\mathbf{x}_{K-1}, \mathbf{x}_K, \mathbf{y}_K) p(\mathbf{x}_{K-1:K}|\mathbf{y}_{1:K}; \boldsymbol{\theta}_K) d\mathbf{x}_{K-1:K} \tag{16}$$

To conduct the integration in Equation 16, Bayes rule is used to rewrite the density

$$\begin{aligned} p(\mathbf{x}_{K-1:K}|\mathbf{y}_{1:K}) &= p(\mathbf{x}_{K-1}|\mathbf{x}_K, \mathbf{y}_{1:K-1}) p(\mathbf{x}_K|\mathbf{y}_{1:K}) \\ &= \frac{p(\mathbf{x}_K|\mathbf{x}_{K-1}) p(\mathbf{x}_{K-1}|\mathbf{y}_{1:K-1})}{p(\mathbf{x}_K|\mathbf{y}_{1:K-1})} p(\mathbf{x}_K|\mathbf{y}_{1:K}) \\ &= p(\mathbf{x}_K|\mathbf{x}_{K-1}) p(\mathbf{x}_{K-1}|\mathbf{y}_{1:K-1}) \frac{p(\mathbf{y}_K|\mathbf{x}_K)}{p(\mathbf{y}_K|\mathbf{y}_{1:K-1})}. \end{aligned} \tag{17}$$

where the parameter $\boldsymbol{\theta}_K$ has been dropped from the density expression for the sake of simplicity. Equation 17 suggests that instead of sampling from $p(\mathbf{x}_{K-1:K}|\mathbf{y}_{1:K}; \boldsymbol{\theta}_K)$, we can sample from $p(\mathbf{x}_K|\mathbf{x}_{K-1}; \boldsymbol{\theta}_K) p(\mathbf{x}_{K-1}|\mathbf{y}_{1:K-1}; \boldsymbol{\theta}_K)$ and each particle should be weighted with the corresponding unnormalized importance weight proportional to $p(\mathbf{y}_K|\mathbf{x}_K; \boldsymbol{\theta}_K)$. Finally, they should be normalized to sum one. This procedure is similar to the sampling performed in the bootstrap filter (Gordon et al., 1993; Arulampalam et al., 2002) in which the particles from the marginalized posterior density at a previous time $p(\mathbf{x}_{K-1}|\mathbf{y}_{1:K-1}; \boldsymbol{\theta}_K)$ evolve forward through the transition density, $p(\mathbf{x}_K|\mathbf{x}_{K-1}; \boldsymbol{\theta}_K)$. However, here we conduct a *single* time step with the only purpose of approximating the expectation (to estimate the parameters) so that no resampling is required. We assume that a particle representation of the filtering posterior density, $p(\mathbf{x}_k|\mathbf{y}_{1:k}; \boldsymbol{\theta}_k)$, is available, which is the case for sequential Monte Carlo techniques and EnKFs. In this work, we use the perturbed observation EnKF (Burgers et al., 1998) and the variational mapping particle filter (Pulido and van Leeuwen, 2018) to infer the marginalized posterior density. The sampling procedure to evaluate Equation 16 is independent of the filtering algorithm and is thus expected to be suitable for any variant of the ensemble Kalman or particle filters.

In this way, we can sample at $K-1$ using the filtering density and then obtain a sample of the forecast at time K through the transition density. The sampling points are denoted as

$$\{\mathbf{x}_{K-1}^{a(j)}\}_{j=1}^{N_p} \sim p(\mathbf{x}_{K-1}|\mathbf{y}_{1:K-1}; \boldsymbol{\theta}_K),$$

and for each j

$$\{\mathbf{x}_K^{f(j,l)}\}_{l=1}^{M_p} \sim p(\mathbf{x}_K|\mathbf{x}_{K-1}^{a(j)}; \boldsymbol{\theta}_K),$$

where the superscripts a and f stand for analysis and forecast. N_p and M_p are the number of particles used to approximate each integral. The number of particles provided by the filter is N_p and for each of these a sample of size M_p is drawn using the transition density. Therefore, we can approximate Equation 16 via Monte Carlo integration as a double sum:

$$\mathcal{E}\left[s(\mathbf{x}_{K-1}, \mathbf{x}_K, \mathbf{y}_K)|\mathbf{y}_{1:K}; \boldsymbol{\theta}_K\right] \approx \sum_{j=1}^{N_p} \sum_{l=1}^{M_p} w_{j,l}\, s(\mathbf{x}_{K-1}^{a(j)}, \mathbf{x}_K^{f(j,l)}, \mathbf{y}_K), \tag{18}$$

where the importance weights $w_{j,l}$ are the normalized counterparts of the unnormalized weights,

$$\bar{w}_{j,l} = p(\mathbf{y}_K|\mathbf{x}_K^{f(j,l)}).$$

In cases in which the number of particles is large, M_p could be chosen equal to 1 producing from each particle at $\mathbf{x}_{K-1}^{a(j)}$ a forecast with a single realization of model error. In this case a single sum remains in Equation 18.

2.3.2 One-step smoother

An alternative to importance sampling is to directly sample from $p(\mathbf{x}_{K-1:K}|\mathbf{y}_{1:K})$ assuming parametric statistics. In order to do this, we need to compute or approximate the smoothing density $p(\mathbf{x}_{K-1}|\mathbf{y}_{1:K})$. In the context of filtering with an EnKF, this can be done using the ensemble variant of the RTS Kalman smoother (EnKS) (Rauch et al., 1965; Cosme et al., 2012). This is a backwards algorithm that is usually applied to get the smoothing densities for the state at all times, but here we are interested in one of them, namely the smoothing density at time $K-1$, thus we only need to do one step backwards of the algorithm. On the other hand, the density of the last state conditioned on all observations is the filtering density that is provided by the EnKF. Hence, if we have

$$\{\mathbf{x}_{K-1}^{s(j)}\}_{j=1}^{N_p} \sim p(\mathbf{x}_{K-1}|\mathbf{y}_{1:K};\boldsymbol{\theta}_K),$$
$$\mathbf{x}_K^{a(j)} \sim p(\mathbf{x}_K|\mathbf{y}_{1:K};\boldsymbol{\theta}_K),$$

we can approximate the expectation in Equation 16 using Monte Carlo integration as

$$\mathcal{E}\left[s(\mathbf{x}_{K-1},\mathbf{x}_K,\mathbf{y}_K)|\mathbf{y}_{1:K};\boldsymbol{\theta}_K\right] \approx \frac{1}{N_p}\sum_{j=1}^{N_p} s(\mathbf{x}_{K-1}^{s(j)},\mathbf{x}_K^{a(j)},\mathbf{y}_K).$$

We generate the EnKS smoothed particles $\mathbf{x}_{K-1}^{s(j)}$ following Cosme et al. (2012) through the equations

$$\mathbf{K}_{K-1}^s = \mathbf{S}_{K-1}^a[(\mathbf{S}_K^f)^T\mathbf{S}_K^f]^{-1}(\mathbf{S}_K^f)^T,$$
$$\mathbf{x}_{K-1}^{s(j)} = \mathbf{x}_{K-1}^{a(j)} + \mathbf{K}_{K-1}^s(\mathbf{x}_K^{s(j)} - \mathbf{x}_K^{f(j)}),$$

where $\mathbf{S}_{K-1}^a$ and $\mathbf{S}_K^f$ consists of particles of the analysis at time $K-1$ and of the forecast particles at time K respectively, each particle centered at their mean, i.e. $\mathbf{x}_K^{a(j)} - \overline{\mathbf{x}}_K^a$ and $\mathbf{x}_K^{f(j)} - \overline{\mathbf{x}}_K^f$, and set as columns. Also, note that the smoothing distribution of the last state $\mathbf{x}_K$ coincides with the filtering distribution, $\mathbf{x}_K^{s(j)} = \mathbf{x}_K^{a(j)}$.

This implementation of an online EM algorithm for the Gaussian case based on the EnKF has some resemblance to the technique proposed in Berry and Sauer (2013). That work also proposes an exponential moving average for the parameters. However, the updates of the parameters are derived from analyzing the cross-correlation, using different time lags, of the forecast based innovations $\boldsymbol{\epsilon}_k = \mathbf{y}_k - \mathbf{H}\mathbf{x}_k^f$ (where $\mathbf{H}$ is the linealization of the observational operator $\mathcal{H}$) which can be related to the matrices $\mathbf{Q}$ and $\mathbf{R}$ once they have reached a stationary regime. On the other hand, our approach is based on likelihood maximization. At evaluation of the sufficient statistics, it makes use of filtered and smoothed estimates of the state. The expectation on Equation 15 is conditioned on all available observations including the observation at time K.

3 Experiments and results

A series of twin experiments was conducted using a known model error covariance, referred as true model error covariance, to evaluate the convergence of the estimator. The chaotic dynamical systems used in the experiments are the Lorenz-63, the one-scale Lorenz-96 and the two-scale Lorenz-96 systems for which equations, parametrizations and implementation details are given in Appendix A. The proposed EM algorithm was implemented with two filters, an ensemble Kalman filter that uses perturbed observations (Burgers et al., 1998) and a particle filter based on gradient flows, the variational mapping particle filter (VMPF, Pulido and van Leeuwen, 2018). Details of the two filters are given in Appendix B. For the proposed importance sampling methodology (Section 2.3.1), we conducted a set of experiments using the variational mapping particle filter (referred to as IS-VMPF) and another one using the ensemble Kalman Filter (IS-EnKF). The set of experiments with the one-step smoother and the EnKF is referred to as OSS-EnKF.

Details of the experiments performed to evaluate the methodology are: the assimilation cycle length, time between observations, is $\Delta t = 0.05$ time units in every experiment. This corresponds to 5 integration time steps for the Lorenz-63 (for which the integration time step is $\delta t = 0.01$) and 50 integration time steps for the Lorenz-96 and the two-scale Lorenz-96 system ($\delta t = 0.001$). We used $N_p = 50$ particles for the filtering techniques in experiments with the Lorenz-63 and the 8 variables Lorenz-96, and $N_p = 100$ for the experiment with the 40 variables Lorenz-96 system. For the importance-sampling-based techniques, we used $M_p = 20$ as the number of samples to compute the Monte Carlo integration in all experiments except for the 40 variables Lorenz-96 for which we used $M_p = 100$. We use the notation $\mathbf{I}_n$ to denote the $n \times n$ identity matrix, N_x the state dimension and N_y the dimension of the observation space.

We chose step sizes of the form $\gamma_k = k^{-\alpha}$ where k indexes the cycle and $\alpha \in (0.5, 1)$ is a tunable parameter. With this scheme, more weight is given to the current estimates in the initial iterations and more weight is given to previous estimates in later cycles when the parameter is expected to be already better estimated. The parameter α controls how is the weight distributed between the current and previous estimations. We chose $\alpha = 0.6$ for all experiments except stated otherwise. This choice is based on manual tuning experiments which evaluate the performance for different values of α.

3.1 Lorenz-63 system

The observations are generated from the 3-dimensional Lorenz-63 dynamical system with Gaussian model and observational uncertainties using diagonal covariance matrices $\mathbf{Q} = \sigma_Q^2 \mathbf{I}_{N_x}$ and $\mathbf{R} = \sigma_R^2 \mathbf{I}_{N_y}$ where $\sigma_Q^2 = 0.3$ and $\sigma_R^2 = 0.5$. We estimate $\mathbf{Q}$ and assume $\mathbf{R}$ is known. We assume the full state is observed which implies that $N_y = N_x$. The uncertainty in the estimator is empirically evaluated by repeating the experiment 30 times with different initial samples and observations.

Figure 1 shows the estimations of the different realizations of the Lorenz-63 experiment for the IS-EnKF, OSS-EnKF and IS-VMPF implementations. Estimations for different initial guesses converge consistently. The estimated mean value is shown with a black line and a slight underestimation of the true value can be detected. The realizations show the convergence of the estimation's uncertainty which occurs after about 500 cycles. This uncertainty also includes sampling noise due to the particle representation of the state distribution.

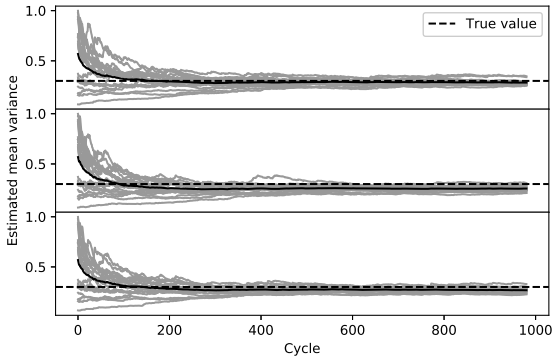

Figure 1: Estimated mean model error variance with different first guesses for the three different variations of the estimation algorithm, (a): IS-EnKF, (b): OSS-EnKF, (c): IS-VMPF

Another experiment was conducted to evaluate the influence of the step size, γ_k, on the convergence. The experiment was repeated with the same observations but the step size changing according to $\gamma_k = k^{-\alpha}$ (LeGland and Mevel, 1997) for different values of α. The impact of different choices for the step size on the estimated model error is shown in Fig. 2. Larger α can be interpreted as more memory in the previous estimates, meaning that in Equation 15, more weight is given to previous model error estimations leading to estimates with less noise. However, they stabilize prematurely overestimating the true value. For smaller step sizes, the resulting estimations are closer to the true value but are noisier. Only results for the IS-VMPF implementation are shown, but results are analogous for IS-EnKF and OSS-EnKF.

3.2 Lorenz-96 system

An experiment with the Lorenz-96 model was conducted in which the observations are generated using a true model error covariance matrix with spatial correlations. We consider positive model error covariance between neighbouring variables (variables next to each other, considering a periodic domain) and null covariance between non-neighbouring ones. The model error covariance between each pair of neighbouring variables is set to 0.09 and the model error variance of each variable is set to 0.3. This model error variance is representative of a two-scale Lorenz-96 model when the small-scale is ignored (Pulido and Rosso, 2017; Pulido et al., 2018). The observational error covariance matrix, $\mathbf{R} = 0.5\mathbf{I}_{N_y}$, is assumed known and only $\mathbf{Q}$ is estimated. The dynamical system used is the 8 variables Lorenz-96 model and the full state is observed.

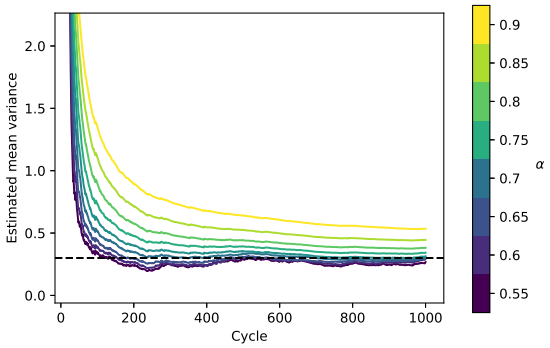

Figure 2: Estimated mean model error variance for different values of α in the step size sequences $\gamma_k = k^{-\alpha}$.

Figure 3 (a) shows the mean of the diagonal, the mean of covariances of neighbouring variables and the mean of covariances of non-neighbouring variables for the OSS-EnKF and IS-VMPF experiments. The mean values are for the only purpose of presentation but note that each matrix entry value is estimated. The major structure of the matrix is correctly recovered. Results for IS-EnKF are not shown but similar results were obtained.

For a meaningful evaluation of the convergence of the method, the full set of observations, $\mathbf{y}_{1:K}$, needs to be considered in the likelihood funtion, i.e. the complete likelihood function. However note that the parameters change in the proposed online method after each observation is assimilated. To find an approximation of the complete likelihood function, we computed state estimates with a complete pass of an EnKF through all the observations, $\mathbf{y}_{1:K}$, using a model error covariance that is fixed in the whole time interval $1{:}K$. With the produced state estimates, we computed an approximation of the loglikelihood. This is conducted for each $\mathbf{Q}_k$ (estimation of $\mathbf{Q}$ at the k-th cycle) of the error covariance. In a similar way, the RMSE with respect to the true state is determined. In Fig. 3 (b), a rapid increase of the loglikelihood and an consistent decrease of RMSE is found. Both quantities then stabilize. This behavior, however, does not guarantee that a global maximum has been reached but that estimations increase the loglikelihood and improve RMSE performance quickly in the first iterations. The RMSE appears to show some improvement even for very long cycles.

The Lorenz-96 experiment is repeated but using a true model error covariance which slowly varies in time with a sigmoid behavior in every non-zero entry value. Figure 4 shows the estimated $\mathbf{Q}$ values. The covariance structure of the matrix is identified and the estimations respond to the slow variation in time of the true parameter. While the matrix increases its values on the non-zero band, the estimations which correspond to values outside this band slightly increase following the behavior of the rest of the matrix. This is in a way compensated by the underestimation of the main diagonal. We obtained similar results for all three implementations but only show them for OSS-EnKF to avoid a cluttered figure. The step-sizes γ_k control the sensitivity of the estimations to past estimations so we expect that the ability of the algorithm to track changes in the error covariances is related on how γ_k is chosen. In particular we expect that if we choose γ_k in such a way that it gives more weight to previous estimations the algorithm will fail to track fast changes in the error covariances. However we do not study this behavior further since it is out of the scope if this work.

In order to assess the performance of the algorithm in a higher-dimensional state space, we conducted an experiment using the 40-variables Lorenz-96 model. Observations of the full state are available every assimilation cycle. We take $\mathbf{Q} = \sigma_Q^2 \mathbf{I}_{N_x}$ and $\mathbf{R} = \sigma_R^2 \mathbf{I}_{N_y}$ with $\sigma_Q^2 = 0.3$ and $\sigma_R^2 = 0.5$.

Figure 5 shows the estimations of the mean of the diagonal and off-diagonal values of the model error covariance as a function of the assimilation cycle. IS-VMPF and OSS-EnKF produce reasonable results, although with a tendency to underestimate. IS-EnKF exhibits the largest underestimation of model error variance. This could in part be explained by the tendency of the EnKF to collapse and because in importance sampling the computation of weights in high dimensions often gives very small unrepresentative values. This underestimation is found for relatively small values of observational error variance, experiments with larger observational errors tend to give better estimates, with slight overestimations as is shown in the next section. The underestimation is not severe enough to induce filter divergence. Additionaly, some slight improvement is found in the three experiments for latter iterations after the fast changes in the first iterations. We note that no inflation factor in the EnKF is used for these experiments. The values outside the diagonal, which correspond to zero, are correctly estimated with every technique.

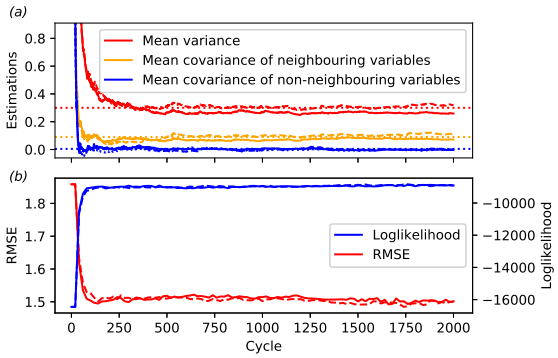

Figure 3: (a): Estimated mean variance (red), mean covariance of neighbouring variables (orange), and mean covariance of non-neighbouring variables (blue) as a function of the assimilation cycle for the OSS-EnKF experiment (solid lines) and the IS-VMPF experiment (dashed lines). True values are shown with dotted lines. (b): Total RMSE (red line) and loglikelihood function (blue line) computed for all the cycles using the corresponding estimated model error.

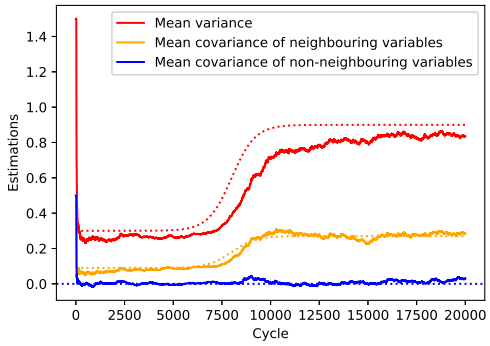

Figure 4: Estimated mean variance (red), mean covariance of neighbouring variables (orange), and mean covariance of non-neighbouring variables (blue) for the OSS-EnKF experiment (solid lines). The corresponding true time varying values are shown with dotted lines.

3.3 Impact and estimation of observation error covariance matrix

The estimation of $\mathbf{Q}$ and $\mathbf{R}$ are closely related (Tandeo et al., 2018) and often $\mathbf{Q}$ compensates for a badly specified $\mathbf{R}$. In order to assess the influence of having different magnitudes of the observational error variance in the estimation of $\mathbf{Q}$, we conducted two experiments:

- *(i)* For an 8-variables Lorenz-96 system, we estimate $\mathbf{Q} = 0.3\mathbf{I}_{N_x}$ using different values for the observational error variance $\mathbf{R} = \sigma_R^2 \mathbf{I}_{N_y}$ and assuming its true value is known in the estimation.

- *(ii)* We repeat experiment *(i)* but instead of using the true observational errors we jointly estimate $\mathbf{Q}$ and $\mathbf{R}$.

For both experiments, we consider a fully observed state and use the OSS-EnKF technique.

Figure 6 (a) shows estimations of the model error with the Lorenz-96 system for different observational error variances assuming they are known. This corresponds to experiment *(i)*. Larger observational errors lead to larger estimations of model error variances and to a slower convergence in the first cycles. Overall, the technique presents a reasonable estimation for a rather broad range of observational errors. The loglikelihood as a function of model error variance for the different observational error variances is shown in Fig. 6 (b). The increase of observational error deteriorates the conditioning of the loglikelihood function, this is in coherence with Pulido et al. (2018). Furthermore, the maximum

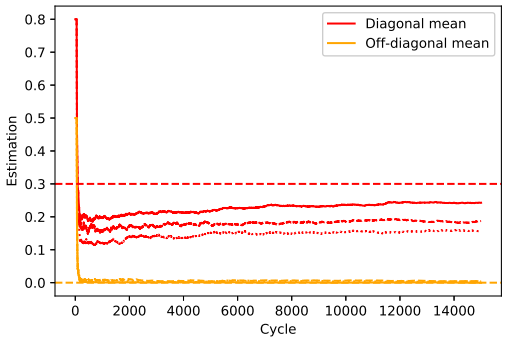

Figure 5: Convergence results for estimations of $\mathbf{Q}$ in the 40-variable Lorenz-96 system for the three different implementations of the proposed algorithm, OSS-EnKF (continuous line), IS-VMPF (dashed line) and IS-EnKF (dotted line).

of the loglikelihood function becomes less well-defined for larger values of σ_R^2. This can be related to the slower convergence of the estimation.

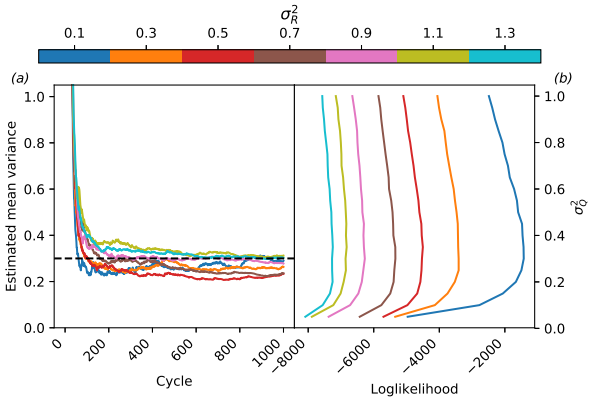

Figure 6: (a): Estimated mean model error variance as a function of time for different observational error variances (continuous lines) using the OSS-EnKF implementation. The true value is indicated with a black dashed line. (b): Approximated loglikelihood as a function of the model error variance for different observational error variances (see color bar).

For experiment *(ii)* (joint $\mathbf{Q}$ and $\mathbf{R}$ estimation), Figs. 7 (a) and (b) show the mean estimated variance of $\mathbf{Q}$ and $\mathbf{R}$ respectively as a function of time. The EM algorithm is able to estimate them jointly and convergence is achieved after about 400 cycles. The quality of the estimation of $\mathbf{Q}$, significantly degrades in comparison with experiment *(i)* which uses the true observational error. The different observational variances are captured rather well but smaller observational error variances are slightly overestimated. The systematic error in the estimation of $\mathbf{Q}$ increases significantly in comparison with the experiment with known observational error. For large observational error variance, the estimations become noisier and the model error variance is overestimated. In the cases that observational error variance is overestimated model error variance is underestimated, so that some compensation between them can be identified.

As was found in experiment *(i)*, the increase of uncertainty in the estimation is expected to be associated with a worse conditioning of the loglikelihood function. Figure 8 shows the contours of the loglikelihood as a function of σ_Q^2 and σ_R^2 for observations generated with two different (true) observational error variances: one to induce a well-conditioned loglikelihood (true values set as $\sigma_Q^2 = 0.3$ and $\sigma_R^2 = 0.5$) and the other to induce an ill-conditioned loglikelihood function ($\sigma_Q^2 = 0.3$ and $\sigma_R^2 = 1.5$). It also shows the means of the diagonals of the estimations of $\mathbf{Q}$ and $\mathbf{R}$ for different repetitions of the experiments changing the initial guess parameters. The likelihood function of the well-

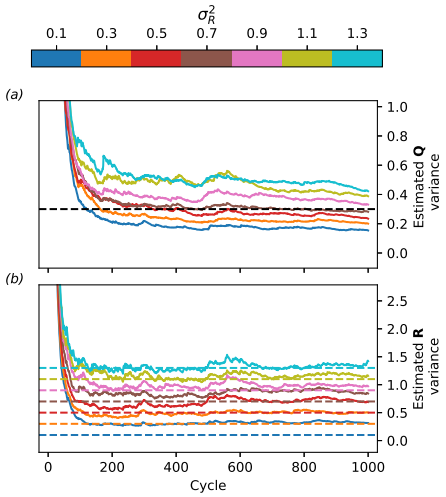

Figure 7: Joint estimates of $\mathbf{Q}$ (top) and $\mathbf{R}$ (bottom) using the OSS-EnKF implementation for experiments with different true observational error variances. The dashed lines indicate the true values.

conditioned experiment has a better defined maximum and the estimations are close to the true value, with a slight tendency to underestimate σ_Q^2. On the other hand, the loglikelihood function of the ill-conditioned experiment has a larger uncertainty in the estimated parameters and there is tendency to overestimate σ_Q^2 and underestimate σ_R^2.

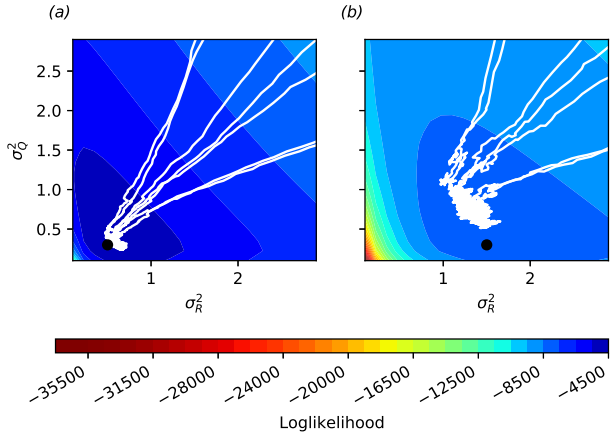

Figure 8: The trajectories of the mean of the diagonal of the estimations for experiments with different initial guess values are shown with white lines. The total loglikelihood function is represented with the colored contours. The true value of the parameters is indicated by the black dot. Panel (a) corresponds to the well-conditioned experiment and (b) to the ill-conditioned one.

3.4 Imperfect model experiment

The experiments described so far have a perfect model structure, the observations are generated with the same model equations as the ones used to produce the forecasts in the filter. The developed methodologies are further assessed in experiments that use an imperfect model. For these experiments, we use a two-scale Lorenz-96 model to generate the observations for which only the large-scale variables are observed. The filter uses a truncated model, a one-scale

Lorenz-96 model, which can only represent the large-scale variables to produce the forecasts. In this more realistic situation, the model error is not necessarily Gaussian and indeed it may have a systematic component. Since we assume the solution from the complete model is not available, model error comes from using a truncated model that ignores the small-scale variables. In this scenario we cannot assess the performance of the proposed methology by comparing the estimations against the true value, as we did in previous experiments. Hence, the quality of the estimations is evaluated with respect to an error covariance matrix which is inferred by comparing the complete and the truncated model forcing terms which serves as a reference to what the model error should represent.

In the truncated model used to perform the estimation, the forcing term of the equations for the variable X_n is F while for the true model, it is $F - \frac{hc}{b}\sum_{j=N_S/N(n-1)+1}^{nN_S/N} Y_j$ (See Appendix A). Computing the covariance of the difference between these forcings, assuming a realization *at each model time step*, we get a covariance matrix structure which serves as a reference to what the model error should represent. Since the online estimations correspond to the overall model error covariance within an assimilation window while this differences correspond to a model time step, a grid search for its best performing scalar multiple was conducted. The performance of this reference matrix in the grid search was evaluated considering each multiple as the model error covariance and computing RMSE and loglikelihood obtained by filtering with an EnKF. The resulting matrix was then used as a reference to compare to the estimations. We take $N = 8$ large-scale variables and $N_S = 256$ small-scale variables. The small-scale variables are completely unobserved and observations of the full large-scale system are taken every 50 model time steps. Observational error covariance is given by $\mathbf{R} = 0.5\mathbf{I}_{N_x}$ as in previous experiments.

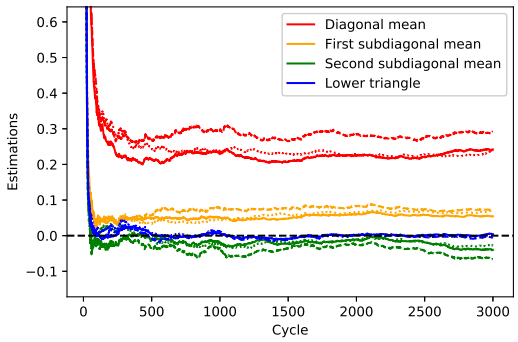

Figure 9: Mean of the estimations of the diagonal, first subdiagonal, second sub diagonal and of the remaining lower triangular matrix for the experiments with a truncated Lorenz-96 model using the OSS-EnKF (solid line), IS-EnKF(dotted line) and IS-VMPF(dashed line) implementations.

Figure 9 shows the estimated model error covariances as a function of time in the imperfect model experiment. The structure of the estimated covariance gives large values on the diagonal, smaller values in the first subdiagonal, negative values in the second subdiagonal and near-zero elsewhere. Overall, there is a rather good agreement between the estimation of the three implementations, IS-EnKF, IS-VMPF, OSS-EnKF. The larger difference in the estimated values between the implementations is found in the variance. The estimated model error variance of this experiment is between 0.2 - 0.3 which is coherent with previous estimations (e.g. Pulido and Rosso, 2017; Scheffler et al., 2019).

Figure 10 (a) shows the full structure of the estimated model error covariance matrix taking the mean of the last 500 OSS-EnKF estimations in the imperfect model experiment. Figure 10 (b) shows the reference matrix obtained offline. The structure of the matrices is similar although entry values of the estimation are lower than the reference. However, note that the offline calculation gives the model error covariance at a single model time step while the estimated one represents the mean model error covariance structure in a whole assimilation cycle.

4 Conclusions

With the goal of developing an effective estimation technique of error covariances based on maximization of the likelihood function for nonlinear hidden Markov models, we proposed an EM-based online scheme that can be coupled with sequential Monte Carlo filtering techniques and ensemble Kalman filters. The main objective was to avoid the use of forward-filtering backward-smoothing strategies since they require to process a batch of observations several times with high storage and computational costs. While the batch EM requires about 30-100 EM iterations (e.g. Pulido

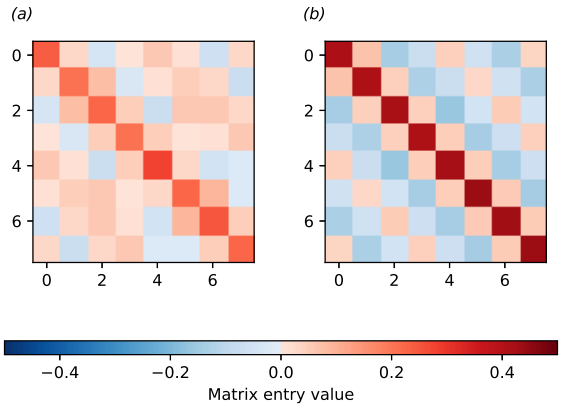

Figure 10: (a): Mean of the last 500 estimates of the model error covariance matrix. (b): Expected covariance matrix obtained from the difference between the exact and the truncated models of the forcing terms.

et al., 2018), each of them imply the application of the smoother for the whole batch of observations, the proposed technique estimates model error covariances on the fly together with the filtering process so that each observation is processed a single time.

The proposed technique shows promising results and is versatile enough to be applied to different data assimilation scenarios while being computationally cheaper than the batch EM. The algorithm can accurately estimate the model error covariance matrix for relatively low-dimensional models with simple error covariance design. Covariance structures with correlation between neighbouring variables were recovered. In the case of model error originated by the use of imperfect model dynamics, the algorithm provided estimates of the covariance matrix with a shape consistent with the problem. Joint estimation of both model and observational error is also feasible. Furthermore, the method can track slow changes in time of the error covariances as shown in the experiments.

A follow-up work will examine the performance in high-dimensional geophysical systems, for which an apriori parameterization of the covariance matrix is required. The computational cost is expected to be feasible: apart from filtering it includes the importance sampling step, the updates of the summary statistics and the parameters. In high-dimensional systems, the scaling impact of the sampling noise in the model error estimates needs to be evaluated. It is expected that localization or regularization through a matrix parameterization are necessary to constrain sampling noise.

The performance varies with the conditioning of the problem, showing overestimation of the parameter when the difference between the scale of model error and observational error is large. Furthermore, the estimates are sensitive to the choice of the step-sizes, γ_k. We think that performance can be improved by incorporating adaptative step-sizes. A potential way to implement this is by reinterpreting the updates of the estimations as a gradient ascent and applying optimization techniques such as adaptive learning rates, e.g. ADADELTA (Zeiler, 2012). This would also circumvent the need of manually tuning the parameter γ_k.

Estimation of model error covariance is key for gradient flow filters and other particle filters which use this information to improve the sampling from the posterior density in high-dimensional state spaces. In particular, the role of the model error covariance is essential in the VMPF. It establishes the prior density as Gaussian mixtures and also has an important role defining the structure of the reproducing kernel Hilbert space in which the particle flow is produced. These experiments are the first ones with this filter in which the structure and values for the model error covariance are not assumed apriori but are determined entirely through the model error covariance estimates. In this sense, the results show there is an important positive feedback between the model error estimation technique and the filter estimates.

Acknowledgments

This work was supported by ANPCyT grant number PICT2015-02368.

References

Tandeo P, Ailliot P, Bocquet M, Carrassi A, Miyoshi T, Pulido M, et al. Joint estimation of model and observation error Covariance matrices in data assimilation: a review. arXiv:180711221 [stat] 2018;.

Houtekamer PL, Mitchell HL. Data assimilation using an ensemble Kalman filter technique. Monthly Weather Review 1998;126(3):796–811.

Anderson JL, Anderson SL. A Monte Carlo implementation of the nonlinear filtering problem to produce ensemble assimilations and forecasts. Monthly Weather Review 1999;127(12):2741–2758.

Miyoshi T. The Gaussian approach to adaptive covariance inflation and its implementation with the local ensemble transform Kalman filter. Monthly Weather Review 2011;139(5):1519–1535.

Chorin AJ, Tu X. Implicit sampling for particle filters. Proceedings of the National Academy of Sciences 2009;106(41):17249–17254.

Atkins E, Morzfeld M, Chorin AJ. Implicit particle methods and their connection with variational data assimilation. Monthly Weather Review 2013;141(6):1786–1803.

Zhu M, van Leeuwen PJ, Amezcua J. Implicit equal-weights particle filter. Quarterly Journal of the Royal Meteorological Society 2016;142(698):1904–1919.

Daum F, Huang J. Nonlinear filters with particle flow induced by log-homotopy. In: Signal Processing, Sensor Fusion, and Target Recognition XVIII, vol. 7336 International Society for Optics and Photonics; 2009. p. 733603.

Pulido M, van Leeuwen PJ. Kernel embedding of maps for sequential Bayesian inference: The variational mapping particle filter. arXiv:180511380 [physics, stat] 2018;.

Snyder C, Bengtsson T, Bickel P, Anderson J. Obstacles to high-dimensional particle filtering. Monthly Weather Review 2008;136(12):4629–4640.

Hamill TM, Whitaker JS, Snyder C. distance-dependent filtering of background error covariance estimates in an ensemble Kalman filter. Monthly Weather Review 2001;129(11):2776–2790.

Ueno G, Higuchi T, Kagimoto T, Hirose N. Maximum likelihood estimation of error covariances in ensemble-based filters and its application to a coupled atmosphere–ocean model. Quarterly Journal of the Royal Meteorological Society 2010;136(650):1316–1343.

Gaspari G, Cohn SE. Construction of correlation functions in two and three dimensions. Quarterly Journal of the Royal Meteorological Society 1999;125(554):723–757.

Stroud JR, Katzfuss M, Wikle CK. A Bayesian Adaptive Ensemble Kalman Filter for Sequential State and Parameter Estimation. Monthly Weather Review 2018;146(1):373–386.

Shumway RH, Stoffer DS. An approach to time series smoothing and forecasting using the EM Algorithm. Journal of Time Series Analysis 1982;3(4):253–264.

Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society: Series B (Methodological) 1977;39(1):1–22.

Tandeo P, Pulido M, Lott F. Offline parameter estimation using EnKF and maximum likelihood error covariance estimates: Application to a subgrid-scale orography parametrization. Quarterly Journal of the Royal Meteorological Society 2015;141(687):383–395.

Dreano D, Tandeo P, Pulido M, Ait-El-Fquih B, Chonavel T, Hoteit I. Estimating model-error covariances in nonlinear state-space models using Kalman smoothing and the expectation-maximization algorithm. Quarterly Journal of the Royal Meteorological Society 2017;143(705):1877–1885.

Pulido M, Tandeo P, Bocquet M, Carrassi A, Lucini M. Stochastic parameterization identification using ensemble Kalman filtering combined with expectation-maximization and Newton-Raphson maximum likelihood methods. Tellus A: Dynamic Meteorology and Oceanography 2018;70(1):1–17.

Lucini M, van Leeuwen PJ, Pulido M. Model uncertainty estimation using the expectation maximization algorithm and a particle flow filter. arXiv:191101511 [statCO] 2019;.

Cappé O. Online sequential Monte Carlo EM algorithm. IEEE Workshop on Statistical Signal Processing Proceedings 2009;p. 37–40.

Andrieu C, Doucet A. Online expectation-maximization type algorithms for parameter estimation in general state space models. In: 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03)., vol. 6; 2003. p. VI–69.

Neal RM, Hinton GE. A view of the EM Algorithm that justifies incremental, sparse, and other variants. In: Learning in Graphical Models NATO ASI Series, Dordrecht: Springer Netherlands; 1998.p. 355–368.

Goodfellow I, Bengio Y, Courville A. Deep Learning. The MIT Press; 2016.

Wu C, Jeff F. On the convergence properties of the EM Algorithm. The Annals of Statistics 1983;11(1):95–103.

LeGland F, Mevel L. Recursive estimation in hidden Markov models. Proceedings of the 36th IEEE Conference on Decision and Control 1997;4:3468–3473 vol.4.

Scheffler G, Ruiz J, Pulido M. Inference of stochastic parametrizations for model error treatment using nested ensemble Kalman filters. Quarterly Journal of the Royal Meteorological Society 2019;145(722):2028–2045.

Gordon NJ, Salmond DJ, Smith AFM. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proceedings F Radar and Signal Processing 1993;140(2):107.

Arulampalam MS, Maskell S, Gordon N, Clapp T. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Transactions on Signal Processing 2002;50(2):174–188.

Burgers G, van Leeuwen PJ, Evensen G. Analysis scheme in the ensemble Kalman filter. Monthly Weather Review 1998;126(6):1719–1724.

Rauch HE, Striebel CT, Tung F. Maximum likelihood estimates of linear dynamic systems. AIAA Journal 1965;3(8):1445–1450.

Cosme E, Verron J, Brasseur P, Blum J, Auroux D. Smoothing problems in a Bayesian framework and their linear Gaussian solutions. Monthly Weather Review 2012;140(2):683–695.

Berry T, Sauer T. Adaptive ensemble Kalman filtering of non-linear systems. Tellus A: Dynamic Meteorology and Oceanography 2013;65(1):20331.

Pulido M, Rosso OA. Model selection: using information measures from ordinal symbolic analysis to select model subgrid-scale parameterizations. Journal of the Atmospheric Sciences 2017;74(10):3253–3269.

Zeiler MD. ADADELTA: an adaptive learning rate method. arXiv:12125701 [cs] 2012;.

A Appendix. Description of the dynamical systems

A.1 Lorenz-63 system

This three-variable chaotic model is given by the equations

$$\begin{aligned}\frac{dx}{dt} &= \sigma(x-y),\\ \frac{dy}{dt} &= x(\rho-z)-y,\\ \frac{dz}{dt} &= xy-\beta z.\end{aligned}$$

We use as parameter values, the standard ones: $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$. The integration is performed with a 4th-order Runge-Kutta algorithm and the used timestep is $\delta t = 0.01$.

A.2 Lorenz-96 system

This is an N-variables model given by the equations

$$\frac{dX_n}{dt} = X_{n-1}(X_{n+1} - X_{n-2}) - X_n + F$$

for $n = 1, ...N$.

Periodic boundary conditions $X_{-1} = X_{N-1}$, $X_0 = X_N$, $X_1 = X_{N+1}$ are considered. The forcing is set to $F = 8$ which causes the system to behave chaotically. The integration is performed with a 4th-order Runge-Kutta algorithm and the timestep taken to be $\delta t = 0.001$. This time step is chosen small in order to coincide with the two-scales Lorenz-96 system which needs a finer temporal resolution for stable numerical integration.

A.3 Two-scale Lorenz-96 system

This model considers N large-scale variables given by the equations:

$$\frac{dX_n}{dt} = X_{n-1}(X_{n+1} - X_{n-2}) - X_n + F - \frac{hc}{b} \sum_{j=N_S/N(n-1)+1}^{nN_S/N} Y_j$$

for $n = 1, ...N$ and M small-scale variables defined by

$$\frac{dY_m}{dt} = cbY_{m+1}(Y_{m-1} - Y_{m+2}) - cY_m + \frac{hc}{b} X_{int[(m-1)/N_S/N]+1}$$

for $m = 1, ...N_S$. We assume periodic conditions $X_j = X_{j\%N}$ and $Y_j = X_{j\%N_S}$ and set the parameters to $F = 20$, $h = 1, b = 10, c = 10$. The integration is performed with a 4th-order Runge-Kutta algorithm and the used timestep is $\delta t = 0.001$.

B Appendix. Filtering and smoothing techniques

B.1 EnKF equations

We used here a stochastic version of EnKF based on Burgers et al. (1998). For each time k, the particles$\{\mathbf{x}_k^{a(j)}\}_{j=1}^N$ and $\{\mathbf{x}_k^{f(j)}\}_{j=1}^N$ of the analysis and forecast distribution respectively, are determined from the equations:

$$\begin{aligned}
\mathbf{x}_k^{f(j)} &= \mathcal{M}(\mathbf{x}_{k-1}^{a(j)}) + \boldsymbol{\eta}_k^{(j)} \\
\mathbf{y}_k^{f(j)} &= \mathcal{H}(\mathbf{x}_k^{f(j)}) + \boldsymbol{\nu}_k^{(j)} \\
\mathbf{K}_k &= \hat{\mathbf{P}}_k^f \mathbf{H}^T [\mathbf{H}\hat{\mathbf{P}}_k^f \mathbf{H}^T + \mathbf{R}_k]^{-1} \\
\mathbf{x}_k^{a(j)} &= \mathbf{x}_k^{f(j)} + \mathbf{K}_k(\mathbf{y}_k - \mathbf{y}_k^{f(j)}),
\end{aligned}$$

where $\hat{\mathbf{P}}_k^f$ stands for the sample covariance of the forecast particles $\{\mathbf{x}_k^{f(j)}\}_{j=1}^N$ and $\boldsymbol{\eta}_k^{(j)} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$, $\boldsymbol{\nu}_k^{(j)} \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$.

B.2 VMPF equations

Here we briefly describe the VMPF but refer the reader to Pulido and van Leeuwen (2018) for a thorough description of it. The VMPF "moves" the particles representing the forecast $p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})$, towards the posterior filtering density$p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$, through a gradient flow using a series of small transformations in the directions of the gradient descent of the Kullback-Liebler divergence between the forecast and the posterior densities. We use an index i to indicate each of these transformations, and since we start with the forecast density we associate it with $i = 0$. The algorithm to implement this technique can be sumarized as follows:

Given a particle representation of the posterior density at time k, $\{\mathbf{x}_k^{(j)}\}_{j=1}^{N_p}$, we evolve the particles with the model to obtain a representation of the forecast at time $k + 1$,

$$\mathbf{x}_{k+1,0}^{(j)} = \mathcal{M}(\mathbf{x}_k^{(j)}) + \boldsymbol{\eta}_{k+1}.$$

Then, a sequence of mappings are applied: each particle is moved following

$$\mathbf{x}_{k+1,i+1}^{(j)} = \mathbf{x}_{k+1,i}^{(j)} - \epsilon \nabla \mathcal{D}_{KL}(\mathbf{x}_{k+1,i}^{(j)})$$

until a stopping criterion is met. The evaluation of the gradient of the Kullback-Liebler divergence, $\nabla \mathcal{D}_{KL}$, in each mapping iteration can be approximated by Monte Carlo integration by considering the transformations lie in a reproducing kernel Hilbert space (details in Pulido and van Leeuwen, 2018).