A multi-fidelity ensemble Kalman filter with hyperreduced reduced-order models

Geoff Donoghue[a,*], Masayuki Yano[a]

[a] *University of Toronto, 4925 Dufferin Street, Toronto, Ontario, M5H 5T6, Canada*

Abstract

We present an efficient data assimilation framework for nonlinear dynamical systems that uses multi-fidelity statistical estimates based on a full-order model and a projection-based hyperreduced order model (ROM) that is trained on-the-fly. This framework is particularly applicable to conservation laws in aerodynamics that yield expensive forward models. The formulation comprises the following technical components: (i) an ensemble Kalman filter to tractably handle high-dimensional, strongly nonlinear dynamical models; (ii) multi-fidelity forecast models, where ROM-based coarse fidelities are constructed on-the-fly; and (iii) hyperreduction for the ROM, based on proper orthogonal decomposition and the empirical quadrature procedure, constructed using the ensemble of full order model trajectories. We show that the multi-fidelity statistical estimates based on efficient, on-the-fly construction of the ROM enables rapid and reliable state estimation for practical nonlinear dynamical systems. We demonstrate the effectiveness of our framework to estimate the state of a separated flow around an airfoil by (i) showing that the ROM-based multi-fidelity method is more accurate than the single-fidelity method for comparable computational cost and (ii) showing that multi-fidelity statistics enable the use of a less accurate surrogate when compared against ROM-only filters, at negligible cost.

Keywords: ensemble filtering, nonlinear hyperreduction, multi-fidelity Monte Carlo, data assimilation, uncertainty quantification, aerodynamics

1. Introduction

The goal of data assimilation is to combine an (imperfect) computational model with (sparse and noisy) real-world data to estimate the state of a dynamical system. A common data assimilation method for complex problems is the ensemble Kalman filter (EnKF), which uses an ensemble-based representation of the uncertain state to extend the classical Kalman filter to large-scale, nonlinear dynamical systems [13, 14, 31]. The EnKF has been successfully applied in meteorology, where atmospheric models and weather-station data are combined to forecast weather; however, the application of the EnKF to *engineering systems* faces several challenges. First, accurate data assimilation of complicated, nonlinear, and large-scale dynamical systems may require a large state ensemble ($\mathcal{O}(100)$) and the associated high computational cost may require a dedicated computing

*Corresponding author

Email addresses: geoff.donoghue@mail.utoronto.ca (Geoff Donoghue), masa.yano@utoronto.ca (Masayuki Yano)

facility [26]; the EnKF may be intractable in typical engineering settings with limited computing resources and rapid turnaround time requirements. Second, standard ensemble-size reduction techniques, such as inflation and localization, rely on the prior physical knowledge of the specific dynamical system; inflation methods spread samples about the mean, typically by a heuristic factor based on *a priori* modeling error, while localization methods suppress spurious long-distance correlations by imposing some correlation length on the covariance. While there exist methods to estimate an optimal inflation factor, choosing a correlation length requires *a priori* knowledge of the system [2, 3, 14, 21], which may not be available for novel engineering systems. To apply the EnKF to engineering systems, we need to significantly reduce the computational cost without relying on prior knowledge. In this work, we leverage recent developments in multi-fidelity estimation and model reduction to achieve this goal.

Multi-fidelity (and multi-level) estimation techniques seek to reduce the number of expensive high-fidelity model evaluations for ensemble-based (i.e., Monte-Carlo) estimators by leveraging a correlated and inexpensive low-fidelity model to reduce the variance of the estimator [17, 18, 34]. To this end, we introduce (a family of) lower-fidelity models to accompany the original high-fidelity model, then construct a multi-fidelity estimate by using (i) a small number of high-fidelity model evaluations, (ii) a large number of lower-fidelity model evaluations, and (iii) a control variate formulation [19] that appeals to the correlation between the high-fidelity and lower-fidelity models. To the best of the authors' knowledge, Hoel *et al.* [24] were the first to apply this idea to the EnKF, whereby multi-level Monte-Carlo estimates of the mean and covariance are used to obtain more accurate state estimates at a reduced cost. Their work was subsequently extended to introduce stronger coupling between fidelity estimates to provide more rapid convergence [25, 10] and also to other ensemble-based data assimilation methods, such as the particle filter [28, 20]. While laying formulational and theoretical foundations for multi-fidelity data assimilation, these works focus on simple scalar dynamical systems. Arguably, the true potential of multi-fidelity data assimilation, as applied to large-scale, nonlinear dynamical systems—where the high-fidelity model evaluation is expensive—is yet to be exploited.

As multi-fidelity estimation techniques rely on many evaluations of a lower-fidelity model that is well-correlated with the high-fidelity model, a rapid and accurate lower-fidelity model can significantly improve the performance of the multi-fidelity data assimilation. To this end, we consider projection-based reduce-order models (ROMs) [38, 7]. Outside of the multi-fidelity context, ROMs have been used to enable rapid (real-time) data assimilation using the (standard) Kalman filter [12], 3d and 4d variational data assimilation [4, 29], and the (single-fidelity) EnKF [32, 30, 35, 11, 23, 43]. For each of these methods, the high-fidelity model (i.e., full-order model (FOM)) is used to construct a ROM in the offline stage. Then, in the online stage, a standard single-fidelity data assimilation technique is used with the ROM as the dynamical model to significantly reduce the computational cost. In these ROM-accelerated (but single-level) approaches, the FOM is not directly invoked in the data assimilation stage. Hence, the approach assumes that we can construct a ROM *a priori* in the offline stage that is capable of accurately approximate all possible states that will be encountered in the data assimilation stage; however, training such a ROM may be computationally intractable for large-scale computational models.

An alternative approach to accelerate data assimilation is to use ROMs as the lower-fidelity model(s) in the aforementioned multi-fidelity estimation techniques. The idea of using a ROM-based multi-fidelity estimator has been explored for forward problems by Vidal-Codina *et al.* [42] and for the EnKF by Popov *et al.* [37]. Unlike the single-level ROM-based EnKF [32, 30, 11, 43],

multi-fidelity EnKF of Popov *et al.* use a (small) ensemble of FOM states to construct a ROM in the offline stage, then use the FOM ensemble and a (large) ensemble of ROM states to compute the multi-fidelity mean and covariance estimates used in the EnKF. Popov *et al.* formally derive the multi-fidelity EnKF based on the theory of linear control variates [19] both to find an optimal mapping between the two fidelities and to avoid the issue of non-positive multi-fidelity covariance estimates, which are present in Hoel's formulation [24, 25]. The modification greatly improves the stability of the EnKF and is arguably requisite for practical engineering problems.

In this work, we extend the multi-fidelity EnKF formulation of Popov *et al.* [37] in two regards. First is the extension to dynamical models that exhibit general (non-polynomial) nonlinearities, which necessitates so-called hyperreduction to construct a ROM. Pagani *et al.* [35] employed a hyperreduced ROM in the EnKF, but did not use a multi-level estimation technique, thereby ignoring already computed information from the FOM that could improve the ROM-based statistical estimates. While there exist many hyperreduction methods [5, 9, 8, 15], in this work we use the empirical quadrature procedure (EQP) [45]. Our second point of departure from Popov *et al.* is the use of on-the-fly model reduction and conditional ROM retraining, which leverages the ensemble of FOM states generated by the EnKF to train ROMs. Namely, we (i) solve for a (small) ensemble of FOM states to be used in the multi-fidelity EnKF; (ii) build a hyperreduced ROM on-the-fly using the FOM state ensemble, conditional on the value of an *a posteriori* error estimate; (iii) solve for a (large) ensemble of ROM solutions; and (iv) combine the FOM and ROM ensembles to compute the multi-fidelity mean and covariance estimates for the EnKF. The on-the-fly construction eliminates the need to anticipate all possible states that might be encountered in the data assimilation process and construct a ROM *a priori* in the offline stage that works for all possible cases. We can instead build and update ROMs that are required for the particular data assimilation problem on-the-fly, only when the ROM quality is estimated to be below some tolerance, reducing the ROM construction costs significantly. Moreover, the FOM solves in the multi-fidelity EnKF naturally yield the training data with which (hyperreduced) ROM is constructed; since this data must be computed to train the ROM the multi-fidelity statistical estimates are hence "free" for the on-the-fly approach. On-the-fly model reduction has been exploited in optimization (e.g., [46, 47]) but, to the best of the authors' knowledge, not in data assimilation.

We summarize the three-fold contributions of this paper: (i) we extend the multi-fidelity EnKF method of [37] to incorporate hyperreduced ROMs to treat general nonlinear dynamical models; (ii) we develop a conditional, on-the-fly ROM training procedure that both ensures the accuracy of the ROM and reduces unnecessary training cost; and (iii) we assess the effectiveness of the method using a separated compressible Navier-Stokes flow past a NACA0012 airfoil. We demonstrate that (a) even with a small FOM ensemble that would result in a significant statistical error, an accurate ROM can be constructed on-the-fly; (b) even if a ROM is not accurate enough to be used in a single-level EnKF by itself, the ROM can improve the state estimate in the multi-fidelity EnKF; and (c) ROMs can be efficiently constructed on-the-fly from the existing FOM states and an efficient hyperreduction method.

2. Multi-fidelity ensemble Kalman filter based on hyperreduced ROMs

2.1. Problem statement

We first provide a mathematical description of a "true" dynamical system whose state we wish to estimate. To this end, we introduce a time interval $I \equiv (0, T]$ and an (infinite-dimensional)

Hilbert space $\mathcal{U}$, in which the true state $u^{\text{true}} : I \to \mathcal{U}$ lies at any time $t \in I$. We assume that the "true" state u^{true} is governed by

$$\begin{aligned} \mathcal{M}^{\text{true}} \frac{\partial u^{\text{true}}}{\partial t}(t) + r^{\text{true}}(u^{\text{true}}(t)) = 0, \quad t \in I, \\ u^{\text{true}}(t=0) = u^{\text{true},0} \in \mathcal{U}, \end{aligned} \tag{1}$$

where $\mathcal{M}^{\text{true}} : \mathcal{U} \to \mathcal{U}'$ is the "true" mass operator (which maps the state variables to the conservative variables), $r^{\text{true}} : \mathcal{U} \to \mathcal{U}'$ is a "true" steady residual operator, $u^{\text{true},0} \in \mathcal{U}$ is the "true" initial condition, and $\mathcal{U}'$ denotes the dual space of $\mathcal{U}$. In general, we have neither the complete knowledge of $\mathcal{M}^{\text{true}}$, $r^{\text{true}}(\cdot)$, and $u^{\text{true},0}$, nor the computational resources to solve the infinite-dimensional problem, and hence it is impossible to solve for u^{true}.

Our goal in data assimilation is to estimate u^{true} based on (imperfect) computational model and experimental/real-world observation data. A computational model incorporates our best knowledge of the (unknowable and uncomputable) "true" dynamical system. We focus on the case where the dynamics are approximated by unsteady nonlinear partial differential equations (PDEs). For instance, if we wish to estimate the state of a fluid dynamics system, then our (continuous) dynamical model may be based on the Navier-Stokes equations with appropriate boundary conditions. We then discretize this continuous model to obtain a computational model. Specifically, we introduce some n-dimensional approximation space $\mathcal{U}_n$ and then seek an approximation to u^{true} as follows: find (model) state $u_n : I \to \mathcal{U}_n$ such that

$$\mathcal{M}_n \frac{du_n}{dt}(t) + r_n(u_n(t)) = 0 \quad \forall\, t \in I, \tag{2}$$

where $\mathcal{M}_n : \mathcal{U}_n \to \mathcal{U}_n'$ is the mass matrix, and $r_n : \mathcal{U}_n \to \mathcal{U}_n'$ is a discretized steady residual operator. The dimension n is of order 10^5 to 10^8 for fluid dynamics systems of engineering interest. As discussed in the Introduction, we refer to this model as the full-order model (FOM).

To facilitate the presentation of the data assimilation algorithm, we also introduce a forecast operator associated with the FOM (2). To this end, we assume that observations are made at K time instances $\{t^k\}_{k=0}^K$ such that $0 = t^0 < t^1 < \cdots < t^K \leq T$. We then introduce a FOM propagation operator $G_n^k : \mathcal{U}_n \to \mathcal{U}_n$ such that

$$u_n^k = G_n^k(u_n^{k-1}) \in \mathcal{U}, \quad k = 1, \ldots, K, \tag{3}$$

where $u_n^{k-1} \equiv u_n(t^{k-1})$ and $u_n^k \equiv u_n(t^k)$ satisfy (2) for $t \in [t^{k-1}, t^k]$. In practice, the application of G_n^k requires the application of a suitable time-marching scheme from t^{k-1} to t^k.

We next introduce observations (or measurements) and an associated observation model. We first introduce an observation space $\mathcal{Y}$; in many cases, an observation is associated with a set of n_{o} discrete sensor measurements and hence $\mathcal{Y} \subset \mathbb{R}^{n_{\text{o}}}$. We next introduce a "true" observation operator $H^{\text{true}} : \mathcal{U} \to \mathcal{Y}$, which yields a set of measurements associated with a given (instantaneous) "true" state. The observed data associated with time instances $\{t^k\}_{k=1}^K$ are

$$y^{\text{obs},k} = H^{\text{true}}(u^{\text{true},k}) + \epsilon, \quad k = 1, \ldots, K, \tag{4}$$

where $\epsilon \sim \mathcal{N}(0, \Gamma)$ is the observation noise with a covariance $\Gamma : \mathcal{Y}' \to \mathcal{Y}$. (Note: while we could absorb the noise ϵ in $H^{\text{true}}(\cdot)$, we treat it separately following standard conventions.) The set $\{y^{\text{obs},k} \in \mathcal{Y}\}_{k=1}^K$ forms the observation data. The "true" observation operator $H^{\text{true}}(\cdot)$ depends on

the precise specification of the observation system used and, just like the "true" dynamical model $r^{\text{true}}(\cdot)$, is unknowable. We hence introduce a (computable) observation model $H_n : \mathcal{U}_n \to \mathcal{Y}$, which incorporates our best knowledge of the observation system, so that $H_n(u_n^k) \approx H^{\text{true}}(u^{\text{true},k})$ when u_n^k approximates $u^{\text{true},k}$ well.

The goal of data assimilation, or more precisely filtering, is to optimally combine the dynamical model (2) (or equivalently (3)) with the observation data (4) up to a given time t^k to obtain estimate the state $u^{\text{true},k}$. We approach the filtering problem using EnKFs [13, 14, 26], a data assimilation method designed for large-scale filtering problems.

2.2. Single-fidelity ensemble Kalman filter

We now present the (standard) single-fidelity ensemble Kalman filter. By way of preliminaries, we introduce notations for a sample, the sample mean, and the sample covariance used throughout this work. To begin, we introduce a random sample $\omega \equiv \{\omega_j\}_{j=1}^M \subset \Omega$ of size M, where Ω is some sample space from which each ω_j is drawn. Given a random function $v \in L^2(\Omega; \mathcal{V})$ whose realizations belong to space $\mathcal{V}$ (e.g., $\mathcal{V} = \mathcal{U}_n$ or $\mathcal{V} = \mathcal{Y}$), the sample mean $\mathbb{E}_M^\omega[v] \in \mathcal{V}$ based on the ensemble $\{v(\omega_j)\}_{j=1}^M$ is given by

$$\mathbb{E}_M^\omega[v] \coloneqq \frac{1}{M} \sum_{j=1}^M v(\omega_j),$$

where the superscript ω and the subscript M signify that the sample mean is computed from the ensemble ω, which is of size M. Similarly, given random functions $v \in L^2(\Omega; \mathcal{V})$ and $w \in L^2(\Omega; \mathcal{W})$ associated with spaces $\mathcal{V}$ and $\mathcal{W}$, the sample covariance $\mathbb{C}_M^\omega[v, w] : \mathcal{W}' \to \mathcal{V}$ based on the ensembles $\{v(\omega_j)\}_{j=1}^M$ and $\{w(\omega_j)\}_{j=1}^M$ is given by

$$\mathbb{C}_M^\omega[v, w] \coloneqq \frac{1}{M-1} \sum_{j=1}^M (v(\omega_j) - \mathbb{E}_M^\omega[v]) \otimes (w(\omega_j) - \mathbb{E}_M^\omega[w]);$$

here $\otimes$ is an operator such that $(v \otimes w)z = \langle w, z \rangle v \in \mathcal{V}$ for all $v \in \mathcal{V}$, $w \in \mathcal{W}$, and $z \in \mathcal{W}'$, where $\langle \cdot, \cdot \rangle$ denotes the duality pairing. For notational brevity, we use the common abbreviation $\mathbb{C}_M^\omega[v] \coloneqq \mathbb{C}_M^\omega[v, v]$.

We now present the (stochastic) EnKF; it comprises an initialization step, which constructs the state ensemble at $t^{k=0}$, followed by repeated applications of the forecast and analysis steps, which advance the ensemble from t^{k-1} to t^k for $k = 1, \ldots, K$, and update the estimate at the end of each observation time step, respectively.

Initialization. We first introduce a random sample $\omega \equiv \{\omega_j\}_{j=1}^M$ and initialize the associated state ensemble

$$\{u_n^{k=0}(\omega_j) \in \mathcal{U}_n\}_{j=1}^M.$$

Forecast step. In the forecast step, we advance the state ensemble by one observation time step from t^{k-1} to t^k:

$$\overline{u}_n^k(\omega_j) = G_n^k(u_n^{k-1}(\omega_j)), \quad j = 1, \ldots, M, \tag{5}$$

where $G_n^k : \mathcal{U}_n \to \mathcal{U}_n$ is the FOM propagation operator (3). The overline on $\overline{u}_n^k(\omega_j)$ denotes that this is the state estimate at time t^k before incorporating the observation $y^{\text{obs},k}$.

Analysis step. In the analysis step, we first compute the Kalman gain $K_n : \mathcal{Y} \to \mathcal{U}_n$ given by

$$K_n^k \coloneqq \mathbb{C}_M^\omega[\overline{u}_n^k, H_n(\overline{u}_n^k)](\mathbb{C}_M^\omega[H_n(\overline{u}_n^k)] + \Gamma)^{-1}, \tag{6}$$

where we recall $\Gamma : \mathcal{Y}' \to \mathcal{Y}$ is the observation covariance. We next introduce perturbed observations $\{y^k(\omega_j) \equiv y^{\text{obs},k} + \eta(\omega_j)\}_{j=1}^M$ for $\eta(\omega_j) \sim \mathcal{N}(0, \Gamma)$ and update the state ensemble according to

$$u_n^k(\omega_j) = \overline{u}_n^k(\omega_j) + K_n^k(y^k(\omega_j) - H_n(\overline{u}_n^k(\omega_j))), \quad j = 1, \ldots, M. \tag{7}$$

In the analysis step, the ensemble statistics are used to yield a (single) Kalman gain (6), and each member is updated based on its discrepancy with the observed data.

The accuracy and cost of the EnKF depend on the ensemble size M. On one hand, the accuracy and the stability of the filter improves with M as the quality of the sample covariances $\mathbb{C}_M[u_n^k, H_n(u_n^k)]$ and $\mathbb{C}_M[u_n^k, H_n(u_n^k)]$ in the Kalman gain improve, reaching the so-called ideal mean-field limit as $M \to \infty$. On the other hand, in a typical data assimilation problem, the forecast step (5) dominates the overall computational cost, and hence the cost of the filter increases linearly with M. Since the sample covariance converges rather slowly (i.e., as $M^{-1/2}$), a large ensemble size may be required to achieve a desired accuracy, which may render the filter unsuitable for engineering settings with a stringent computational resource limit or a limited turnaround time requirement, as discussed in the Introduction. To overcome the limitations posed by the slow convergence and the expensive forward model, we will consider a multi-fidelity EnKF (MFEnKF).

Multi-fidelity methods, as developed herein, reduce the computational cost required to obtain accurate statistics by using a ROM that can be evaluated much more rapidly than the original model, while also improving the accuracy of these ROM statistical estimators by correcting the ROM-computed statistical estimates with samples collected from the FOM.

2.3. Reduced-order model

We now introduce a projection-based ROM. Our ROM for nonlinear PDEs builds on two ingredients: (i) a reduced basis (RB) and (ii) hyperreduction. An RB $\{\phi_i \in \mathcal{U}_n\}_{i=1}^r$ spans an r-dimensional subspace of $\mathcal{U}_n$ for $r \ll n$ and is designed to provide a rapidly convergent approximation of the solution $u_n : I \to \mathcal{U}_n$ as r increases. We will discuss the construction of an RB in the context of MFEnKF in Section 2.5 and for now assume that $\{\phi_i\}_{i=1}^r$ is given. We also introduce an associated linear RB operator $\Phi : \mathbb{R}^r \to \mathcal{U}_n$ such that $\Phi v_r = \sum_{i=1}^r \phi_i v_{r,i}$ for all $v_r \in \mathbb{R}^r$.

The Galerkin projection of the FOM (2) onto the RB space yields the following (nonhyperreduced) ROM problem: find $\hat{u}_r : [0, T] \to \mathbb{R}^r$ such that

$$\mathcal{M}_r \frac{d\hat{u}_r}{dt}(t) + \Phi^* r_n(\Phi \hat{u}_r(t)) = 0 \quad \forall t \in I,$$

where $\mathcal{M}_r \coloneqq \Phi^* \mathcal{M}_n \Phi \in \mathbb{R}^{r \times r}$ is the ROM mass matrix, and $\Phi^* : \mathcal{U}' \to \mathbb{R}^r$ denotes the formal adjoint of Φ. Note that while we have reduced the dimension of the approximation space from n to $r \ll n$, the evaluation of this ROM is still computationally expensive because the evaluation of the residual $\Phi^* r_n(\Phi \cdot) : \mathbb{R}^r \to \mathbb{R}^r$ requires the evaluation of the FOM residual $r_n : \mathcal{U}_n \to \mathcal{U}_n'$, which requires $\mathcal{O}(n)$ operations.

To accelerate the evaluation of the ROM—particularly for nonlinear problems—we introduce the second ingredient: hyperreduction. We again defer the detailed discussion of the hyperreduction procedure to Section 2.5. For now, we assume that we can construct a hyperreduced ROM operator $r_r : \mathbb{R}^r \to \mathbb{R}^r$ such that $r_r(v_r) \approx \Phi^* r_n(\Phi v_r)$ for all relevant $v_r \in \mathbb{R}^r$. Our hyperreduced ROM problem is as follows: find $u_r : [0, T] \to \mathbb{R}^r$ such that

$$\mathcal{M}_r \frac{du_r}{dt}(t) + r_r(u_r(t)) = 0 \quad \forall t \in I. \tag{8}$$

To facilitate the discussion of the MFEnKF, we also introduce the associated ROM forecast operator $G_r^k : \mathbb{R}^r \to \mathbb{R}^r$ from t^{k-1} to t^k:

$$u_r^k = G_r^k(u_r^{k-1}), \tag{9}$$

where $u_r^{k-1} = u_r(t^{k-1})$ and $u_r^k = u_r(t^k)$ are the solutions of (8) evaluated at times t^{k-1} and t^k, respectively. We also introduce the reduced observation operator $H_r : \mathbb{R}^r \to \mathcal{Y}$ such that

$$H_r(v_r) = H_n(\Phi v_r) \quad \forall v_r \in \mathbb{R}^r.$$

2.4. Multi-fidelity ensemble Kalman filter

We now present the MFEnKF introduced by Popov *et al.* [37]. The goal of the MFEnKF is to reduce the (FOM) ensemble size required to achieve a given accuracy relative to the standard EnKF by using a linear control variate to reduce the variance of the statistical estimates. Namely, we introduce a new random function called the total variate

$$z_n^k = u_n^k - S(u_r^k - \mathbb{E}[u_r^k]), \tag{10}$$

where the FOM state $u_n^k \in L^2(\Omega; \mathcal{U}_n)$ is the principle variate, the ROM state $u_r^k \in L^2(\Omega; \mathbb{R}^r)$ is the control variate, and $S : \mathbb{R}^r \to \mathcal{U}_n$ is the gain operator to be optimized. (Terminology follows Popov *et al.* [37].) The total variate z_n^k has the same mean as the principle variate u_n^k; however, assuming the control variate u_r^k is well correlated with u_n^k and the gain S is appropriately chosen, the total variate z_n^k has a smaller variance than the principle variate u_n^k, and thus will converge more rapidly with the ensemble size than the original estimator. The main idea of the MFEnKF is to apply the EnKF to the total variate z_n^k instead of the principle variate u_n^k to reduce the FOM ensemble size required to achieve a given accuracy. In practice, as $\mathbb{E}[u_r^k]$ is unknown, we approximate it using the ancillary variate, which is another ROM state $\tilde{u}_r^k \in L^2(\Omega; \mathbb{R}^r)$ that is independent of both the principle and control variate. For brevity, we refer to [37] for detailed derivation and present here (only) the formulation.

Similarly to the (single-level) EnKF, the MFEnKF comprises an initialization step, which collects a set of state ensembles at $t^{k=0}$, followed by repeated applications of the forecast and analysis steps, which advance the set of ensembles from t^{k-1} to t^k for $k = 1, \ldots, K$, updating each ensemble member after each observation time step.

Initialization. We first introduce ensembles $\omega \equiv \{\omega_j\}_{j=1}^{M_n}$ and $\tilde{\omega} \equiv \{\tilde{\omega}_j\}_{j=1}^{M_r}$ for $M_r > M_n$ and the associated state ensembles

$$\{u_n^{k=0}(\omega_j) \in \mathcal{U}_n\}_{j=1}^{M_n}, \quad \{u_r^{k=0}(\omega_j) \in \mathbb{R}^n\}_{j=1}^{M_n}, \quad \{u_r^{k=0}(\tilde{\omega}_j) \in \mathbb{R}^n\}_{j=1}^{M_r},$$

where $u_r^{k=0}(\omega_j) = \mathcal{M}_r^{-1}\Phi^*\mathcal{M}_n u_n^{k=0}(\omega_j)$. The first two ensembles are "paired" in the sense that they are associated with the same point in the sample space $\omega_j, j = 1, \ldots, M_n$, while the last ensemble is associated with a different set of samples $\tilde{\omega}$ of size $M_r > M_n$.

Forecast step. In the forecast step, we advance all the state ensembles by one observation time step from t^{k-1} to t^k:

$$\begin{aligned}
\overline{u}_n^k(\omega_j) &= G_n^k(u_n^{k-1}(\omega_j)), \quad j = 1, \ldots, M_n, \\
\overline{u}_r^k(\omega_j) &= G_r^k(u_r^{k-1}(\omega_j)), \quad j = 1, \ldots, M_n, \\
\overline{u}_r^k(\tilde{\omega}_j) &= G_r^k(u_r^{k-1}(\tilde{\omega}_j)), \quad j = 1, \ldots, M_r,
\end{aligned}$$

where $G_n^k : \mathcal{U}_n \to \mathcal{U}_n$ and $G_r^k : \mathbb{R}^r \to \mathbb{R}^r$ are the FOM and ROM propagation operators (3) and (9), respectively. The overlines on the states again denote that they are the state estimates at t^k before incorporating the observation.

Remark 1. *As the ROM approximates the dynamics in a reduced space, the ROM propagated state is in general different from the FOM propagated state projected to the ROM space: i.e., $G_r^k(\Phi^* u^{k-1}) \neq \Phi^* G_n^k(u^{k-1})$. As a result, the control variate $\overline{u}_r^k(\omega_j)$ can "drift" from the principal variate $\overline{u}_n^k(\omega_j)$, reducing the correlation between the variates. To alleviate the issue, we follow [37] and reinitialize the ROM state at the beginning of each forecast step: $u_r^{k-1}(\omega_j) = \mathcal{M}_r^{-1}\Phi^*\mathcal{M}_n u_n^{k-1}(\omega_j)$.*

Analysis step. To describe the multi-fidelity analysis procedure, we first introduce the multi-fidelity counterparts of the standard single-fidelity covariance estimates. The multi-fidelity measurement covariance—the covariance of the total variate of the measurement prediction—is given by

$$\begin{aligned}\mathbb{C}^{\omega,\tilde{\omega}}_{M_n,M_r}[H(\overline{u}^k)] \coloneqq &\mathbb{C}^{\omega}_{M_n}[H_n(\overline{u}_n^k)] + \frac{1}{4}\mathbb{C}^{\omega}_{M_n}[H_r(\overline{u}_r^k)] - \frac{1}{2}\mathbb{C}^{\omega}_{M_n}[H_n(u_n^k), H_r(\overline{u}_r^k)] \\ &- \frac{1}{2}\mathbb{C}^{\omega}_{M_n}[H_n(\overline{u}_r^k), H_r(u_n^k)] + \frac{1}{4}\mathbb{C}^{\tilde{\omega}}_{M_r}[H_r(\overline{u}_r^k)],\end{aligned}$$

the multi-fidelity state-measurement covariance is similarly given by

$$\begin{aligned}\mathbb{C}^{\omega,\tilde{\omega}}_{M_n,M_r}[\overline{u}^k, H(\overline{u}^k)] \coloneqq &\mathbb{C}^{\omega}_{M_n}[\overline{u}_n^k, H_n(\overline{u}_n^k)] + \frac{1}{4}\mathbb{C}^{\omega}_{M_n}[\Phi\overline{u}_r^k, H_r(\overline{u}_r^k)] - \frac{1}{2}\mathbb{C}^{\omega}_{M_n}[\overline{u}_n^k, H_r(\overline{u}_r^k)] \\ &- \frac{1}{2}\mathbb{C}^{\omega}_{M_n}[\Phi\overline{u}_r^k, H_r(u_n^k)] + \frac{1}{4}\mathbb{C}^{\tilde{\omega}}_{M_r}[\Phi\overline{u}_r^k, H_r(\overline{u}_r^k)],\end{aligned}$$

we refer to Popov *et al.* [37] for the derivation of these statistics. We replace the standard, single-level covariance estimates with their multi-fidelity counterparts to obtain the associated multi-fidelity Kalman gain:

$$K_{\mathrm{MF}}^k \coloneqq \mathbb{C}^{\omega,\tilde{\omega}}_{M_n,M_r}[\overline{u}^k, H(\overline{u}^k)](\mathbb{C}^{\omega,\tilde{\omega}}_{M_n,M_r}[H(\overline{u}^k)] + \tfrac{1}{2}\Gamma)^{-1}.$$

We then introduce sets of perturbed observations $\{y^k(\omega_j) \equiv y^{\mathrm{obs},k} + \eta(\omega_j)\}_{j=1}^{M_n}$ for $\eta(\omega_j) \sim \mathcal{N}(0,\Gamma)$ and $\{y^k(\tilde{\omega}_j) \equiv y^{\mathrm{obs},k} + \eta(\tilde{\omega}_j)\}_{j=1}^{M_r}$ for $\eta(\tilde{\omega}_j) \sim \mathcal{N}(0,\Gamma)$. Finally, we update the state ensembles:

$$u_n^k(\omega_j) = \overline{u}_n^k + K_{\mathrm{MF}}^k(y^k(\omega_j) - H_n(\overline{u}_n^k(\omega_j))), \quad j = 1,\dots,M_n, \tag{11}$$

$$u_r^k(\omega_j) = \overline{u}_r^k + K_{\mathrm{MF}}^k(y^k(\omega_j) - H_n(\overline{u}_r^k(\omega_j))), \quad j = 1,\dots,M_n, \tag{12}$$

$$u_r^k(\tilde{\omega}_j) = \overline{u}_r^k + K_{\mathrm{MF}}^k(y^k(\tilde{\omega}_j) - H_n(\overline{u}_r^k(\tilde{\omega}_j))), \quad j = 1,\dots,M_r. \tag{13}$$

While not strictly required in the forecast or analysis steps, we may also compute the multi-fidelity estimate of the mean as

$$\mathbb{E}^{\omega,\tilde{\omega}}_{M_n,M_r}[u] = \mathbb{E}^{\omega}_{M_n}[u_n] - \frac{1}{2}\Phi\mathbb{E}^{\omega}_{M_n}[u_r] + \frac{1}{2}\Phi\mathbb{E}^{\tilde{\omega}}_{M_r}[u_r] \tag{14}$$

at any time $t \in I$, which serves as our estimate of u^{true}.

The accuracy of the MFEnKF depends on the FOM ensemble size M_n, the ROM ensemble size M_r, and how accurately the ROM G_r^k approximates the FOM G_n^k. The cost of the MFEnKF

depends on M_n, M_r, the FOM evaluation cost, the ROM evaluation cost, and the ROM training cost, which itself depends on M_n. We will numerically study how these parameters affect the performance of the MFEnKF in Section 3.

Before we conclude the discussion of MFEnKF, we note a difference between our formulation and that of Popov *et al.* [37] in the treatment of multi-fidelity noise terms:

Remark 2. *When considering a multi-fidelity implementation of the noise terms, Popov et al. propose two different methods for constructing with multi-fidelity noise terms: (i) "total variate uncertainty consistency", which assumes the noise represents the uncertainty in the total variate, yielding a noise covariance in the ancillary variate that is triple the noise covariance of the principal and control variates; and (ii) "control space uncertainty consistency", which assumes the operator for the measurement of the control and ancillary variates are the same and therefore have the same noise covariance, yielding a measurement covariance that is scaled by a factor of* $1/2$ *compared to the standard EnKF. Popov et al. use method (i); we have tried both approaches and found a slight performance increase using method (ii) for our model problems, and hence we use method (ii).*

Having discussed the MFEnKF based on a ROM, we now discuss on-the-fly and conditional training of the ROM in Sections 2.5 and 2.6, respectively.

2.5. On-the-fly ROM training in the MFEnKF

In this work we construct the ROM on-the-fly instead of using a more traditional offline-online computational decomposition. As discussed in the Introduction, on-the-fly training provides two distinct advantages compared to the "standard" offline-online computational decomposition in the context of the MFEnKF. First, it eliminates the need to construct a general ROM that works for all potential states and enables the generation of a more specialized ROM that is tailored for the specific solution trajectories that are relevant to the particular instance of the MFEnKF; we expect the more specialized, and hence lower-dimensional, ROM to require a smaller training set and also enable more rapid evaluation. Second, the FOM solves in the MFEnKF naturally yield the training states, which we can use to construct the (hyperreduced) ROM; the cost associated with computing the multi-fidelity statistics in the offline stage the offline-online approach is hence practically negligible for the on-the-fly approach.

As discussed in Section 2.3, our (hyperreduced) ROM builds on two ingredients: a RB space and hyperreduction. We now describe our approach to construct these two ingredients on-the-fly in the MFEnKF. Suppose we wish to construct a ROM at time t^k. We first collect all M_n FOM trajectories generated by the MFEnKF up to that time to form a training set:

$$U^k_{\text{train}} \equiv \{\{u_n(\omega_j; t)\}_{t\in\tau}\}_{j=1}^{M_n}, \tag{15}$$

where $\tau = [t^\star, t^k]$ for $t^\star \in [0, t^{k-1}]$ is a set of time instances used by the underlying time integrator. (In general, a single observation time interval $[t^k, t^{k-1}]$ comprises multiple time integrator steps.) We then apply proper orthogonal decomposition (POD) to the training set to obtain the r-dimensional RB:

$$\Phi = \text{POD}_r(U_{\text{train}}).$$

In other words, the RB is constructed from all trajectories obtained by the FOM ensemble of size M_n since time $t^\star$. Longer trajectories and larger ensemble sizes typically improve the ROM quality, but increase the training costs. We further control this ROM accuracy/cost tradeoff by finding

only r reduced basis functions such that the eigenvalue corresponding to the r-th RB function is a factor δ_{RB} smaller than the largest eigenvalue. We will show in Section 3 that even with a very loose δ_{RB} the MFEnKF can provide an accurate state estimate.

We now describe our hyperreduction procedure. In this work, we use an approach based on a reduced quadrature (RQ) constructed using the empirical quadrature procedure (EQP) [45]. To this end, we assume that the FOM residual $r_n(\cdot)$ is associated with a variational (or weak) formulation, such as the finite element method. These methods require the evaluation of integrals over the spatial domain and potentially element facets, which are approximated using appropriate quadrature rules in practice. Quadrature rules yield a natural decomposition of the residual $r_n(\cdot)$ in terms of individual quadrature-point-wise residuals $r_{n,q}(\cdot)$:

$$r_n(\cdot) = \sum_{q=1}^{Q_n} \rho_q r_{n,q}(\cdot),$$

where $\{\rho_q\}_{q=1}^{Q_n}$ is the set of quadrature weights, and Q_n is the number of quadrature points, which is $\mathcal{O}(n)$. The main idea of EQP is to replace the original quadrature weights $\{\rho_q\}_{q=1}^{Q_n}$ with a sparse set of quadrature weights $\{\rho_q^\star\}_{q=1}^{Q_n}$, which are mostly zero, with $\|\rho_q^\star\|_0 = Q_r$, and to construct a hyperreduced ROM residual operator

$$r_r(\cdot) = \sum_{q=1}^{Q_n} \rho_q^\star \Phi^* r_{n,q}(\Phi\ \cdot\ ) \approx \Phi^* r_n(\Phi\ \cdot\ ).$$

The choice of the sparse quadrature weights $\{\rho_q^\star\}_{q=1}^{Q_n}$ is crucial to ensure the ROM is inexpensive yet accurate.

We find the sparse quadrature weights $\{\rho_q^\star\}_{q=1}^{Q_n}$ using the EQP. The EQP finds an RQ rule that meets a set of accuracy (or "residual matching") conditions at a set of training states. In the context of the MFEnKF, we use the U_{train}^k given by (15) as the training set, and obtain the following: find $\rho^\star \in \mathbb{R}^Q$ such that

$$\rho^\star = \underset{\hat{\rho}\in\mathbb{R}^{Q_n}}{\arg\min} \|\hat{\rho}\|_0$$

subject to the non-negativity constraint

$$\hat{\rho}_q \geq 0, \quad q = 1, \ldots, Q_n,$$

the constant accuracy constraint

$$\left| \sum_{q=1}^{Q_n} \rho_q - \sum_{q=1}^{Q_n} \hat{\rho}_q \right| \leq \delta_{\text{RQ}},$$

and the manifold accuracy constraint (i.e., "residual matching condition")

$$|u_{r,i} e_i^T \Phi^* r_n(\Phi u_r) - \sum_{q=1}^{Q_n} \hat{\rho}_q u_{r,i} e_i^T \Phi^* r_{n,q}(\Phi u_r)| \leq \delta_{\text{RQ}}/r, \quad i = 1, \ldots, r, \tag{16}$$

for all ROM training states $u_r \in U_{\text{train},r}^k$; here, $\delta_{\text{RQ}} \in \mathbb{R}_{>0}$ is the EQP tolerance, $u_{r,i}$ denotes the i-th component of $u_r \in \mathbb{R}^r$, $e_i \in \mathbb{R}^r$ is the canonical unit vector, and the ROM training

set is the projection of the FOM training set (15) onto the RB space (i.e., $U^k_{\text{train},r} = \{u_r = \mathcal{M}_r^{-1}\Phi^*\mathcal{M}_n u_n \mid u_n \in U^k_{\text{train},r}\}$). Just as the POD tolerance $\delta_{\text{RB}} \in \mathbb{R}_{>0}$ controls the size of the RB and hence the accuracy of the ROM approximation space, the EQP tolerance $\delta_{\text{RQ}} \in \mathbb{R}_{>0}$ controls the number of the RQ points and hence the accuracy of the RQ residual.

Remark 3. *The EQP is a flexible method to construct a hyperreduced ROM based on an RQ, and the hyperreduction error controlled by the EQP depends on the choice of the manifold accuracy constraints* (16)*. The EQP has been designed to control the H^1-norm of the (steady) solution error [45] and quantity of interest in steady [44] and unsteady [40] problems. In this work, we introduce a different accuracy constraint that is intended to produce a hyperreduced model that is "globally accurate" in the sense that it provides an accurate estimate of the state as well as all of the observed quantities. If the dynamical model is given by a conservation law, one way to approach this problem is to control the (mathematical) entropy, whose control has been shown to result in a global error control in the context of adaptive mesh refinement [16]. For an appropriate variational formulation of a system of conservation laws expressed in entropy variables, the (mathematical) entropy field associated with the state $v_n \in \mathcal{U}_n$ is given by $v_n^T r_n(v_n)$ [22, 27, 6]. In the context of the ROM approximation, the mathematical entropy integrated over the domain associated with the ROM state vector v_r is given by $v_r^T r_r(v_r)$. The EQP accuracy constraint* (16) *is designed to control the error in the mathematical entropy due to hyperreduction so that $v_r^T r_r(v_r) \approx v_r^T \Phi^* r_n(\Phi v_r)$ for relevant ROM states.*

2.6. Conditional ROM training

The goal of on-the-fly ROM training is to minimize the total training time while simultaneously maintaining an accurate ROM surrogate. Following each observation time interval $[t^{k-1}, t^k]$, $k = 1, \ldots, K$, we decide whether the ROM should be recomputed. If the principal (FOM) and control (ROM) variates are well correlated, then we need not reconstruct our RB and RQ. Conversely, if the ROM is inaccurate, we must reconstruct the ROM despite the training costs. To estimate the ROM accuracy, we compute an error indicator

$$\theta \equiv \sqrt{\mathbb{E}^{\omega}_{M_n}\left[\|u_n^k - \Phi u_r^k\|^2_{L^2(D)}\right]} \tag{17}$$

at the end of each control variate forecast $\overline{u}_r^k(\omega_j) = G_r^k(u_r^{k-1}(\omega_j))$, where $\|\cdot\|_{L^2(D)}$ denotes the L^2 norm over the spatial domain. Whenever θ is above some user defined value $\delta_u \in \mathbb{R}_{\geq 0}$, we reconstruct the ROM and rerun the control forecast; otherwise, we keep the previously constructed ROM and run the ancillary forecast $\overline{u}_r^k(\tilde{\omega}_j) = G_r^k(u_r^{k-1}(\tilde{\omega}_j))$.

Algorithm 1 summarizes the MFEnKF, and Figure 1 illustrates the algorithm. The illustration begins on the far left with three ensembles of samples drawn from the initial condition: $\{u_n^{k=0}(\omega_j)\}_{j=1}^{M_n}$, which is represented by large squares; and ensembles $\{u_r^{k=0}(\omega_j)\}_{j=1}^{M_n}$ and $\{u_r^{k=0}(\tilde{\omega}_j)\}_{j=1}^{M_r}$, which are represented together as small circles. In the first forecast step, we first use the FOM operator $G_n^{k=1}(\cdot)$ to propagate the FOM ensemble to t^1, where the first set of observational data $y^{\text{obs},k}$ is collected. We use the ensemble of the FOM solution trajectories from this forward solve to construct the ROM operator $G_r^{k=1}(\cdot)$. We then invoke the ROM to rapidly propagate ROM ensembles to t^1. We then perform the first analysis step: we compute the multi-fidelity statistics using both the ROM and FOM states, compute the multi-fidelity Kalman gain, and incorporate the observations to update all three ensembles. In the subsequent forecast steps,

Algorithm 1 MFEnKF based on hyperreduced ROM

Generate initial states $\{u_n^{k=0}(\omega_j)\}_{j=1}^{M_n}$, $\{u_r^{k=0}(\omega_j)\}_{j=1}^{M_n}$, and $\{u_r^{k=0}(\tilde{\omega}_j)\}_{j=1}^{M_r}$ for $M_r \gg M_n$
for $k = 1, 2, \ldots, K$ **do**
    Forecast step:
    for $j = 1, 2, \ldots, M_n$ **do**
        FOM forecast: $\overline{u}^k(\omega_j) = G_n^k(u^{k-1}(\omega_j))$
    if $k = 1$ **then**
        ROM construction: use POD and EQP on $\{\{u(\omega_j, t)\}_{t\in[0,t^k]}\}_{j=1}^{M_n}$ to build $G_r^k(\cdot)$
    for $j = 1, 2, \ldots, M_n$ **do**
        Control variate ROM forecast: $\overline{u}_r^k(\omega_j) = G_r^k(u_r^{k-1}(\omega_j))$
    if $k \neq 1$ **and** $\theta > \delta_u$ **then**
        ROM reconstruction: use POD and EQP on $\{\{u(\omega_j, t)\}_{t\in[0,t^k]}\}_{j=1}^{M_n}$ to build $G_r^k(\cdot)$
        for $j = 1, 2, \ldots, M_n$ **do**
            Control variate ROM forecast: $\overline{u}_r^k(\omega_j) = G_r^k(u_r^{k-1}(\omega_j))$
    for $j = 1, 2, \ldots, M_r$ **do**
        Ancillary variate ROM forecast: $\overline{u}_r^k(\tilde{\omega}_j) = G_r^k(u_r^{k-1}(\tilde{\omega}_j))$
    Analysis step:
    Compute $\mathbb{E}^{\omega,\tilde{\omega}}_{M_n,M_r}[u], \mathbb{C}^{\omega,\tilde{\omega}}_{M_n,M_r}[H(\overline{u}^k)], \mathbb{C}^{\omega,\tilde{\omega}}_{M_n,M_r}[\overline{u}^k, H(\overline{u}^k)]$
    ML update: compute ML Kalman gain and update using $y^{\text{obs},k}$

we check the accuracy of the ROM using the error indicator θ and, as necessary, reconstruct the ROM before we propagate the ancillary variates (and update the control variates).

We highlight the benefits of the conditional *and* on-the-fly ROM retraining. As stated previously, on-the-fly training offers two distinct benefits relative to offline-trained ROMs: (i) it avoids the need to anticipate all potential states that the ROM might encounter and the associated expensive pre-training of a globally accurate ROM in the offline stage; and (ii) by running in conjunction with the FOM, the ROM training data is enriched, so the ROM can become more accurate while data assimilation proceeds. Offline training often necessitates collecting and training over more data than is necessary, as it is difficult to know *a priori* the required amount of training to achieve a minimally viable surrogate. In addition, conditional training (iii) avoids excessive ROM training by ensuring the RB and RQ are only recomputed when it is deemed necessary to do so. If our algorithm decides that we only need to train over, say, the first two assimilation windows, then the training costs would be significantly smaller than updating the ROM after each assimilation step.

Remark 4. *The effectiveness of the conditional ROM training strongly depends on the nature of the problem. Specifically, if the dominant features are largely unchanged over time, then a ROM trained using the snapshots from an earlier time window can be reused over many time windows. However, if the problem is convection-dominated and the dominant features translate over time, then a frequent ROM update would be required.*

Remark 5. *We refer to the costs of the MFEnKF relative to the (single-level) EnKF that uses ROM as being "free" or practically negligible because (i) the majority of the cost for a multi-fidelity estimator is borne by computing the FOM solutions comprising the finest fidelity in the estimator and (ii) these FOM solutions must be computed anyway in order to train the ROM.*

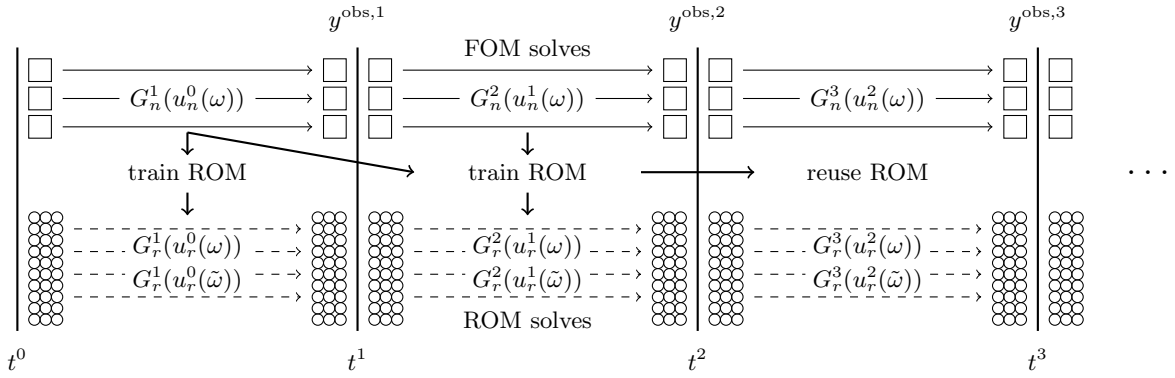


Figure 1: Illustration of the MFEnKF algorithm. Squares denote FOM states, and circles denote ROM states. Each state is propagated from one observation time to the next and then updated by assimilating observational data $y^{\text{obs},k}$ according to (13). The ROM is conditionally trained based on all previous FOM ensemble-temporal solution snapshots since $t^\star$.

Hence, one can replace the single-level Monte Carlo estimator in a single-level ROM-based EnKF with a faster-converging multi-fidelity estimator. Doing so requires only the construction of the multi-fidelity covariance estimator, the cost of which is negligible compared to computing the FOM solutions and training the ROM. Arguably, using a single-fidelity estimator when the training data is available is tantamount to throwing away valuable information that can improve the convergence of the statistical estimator.

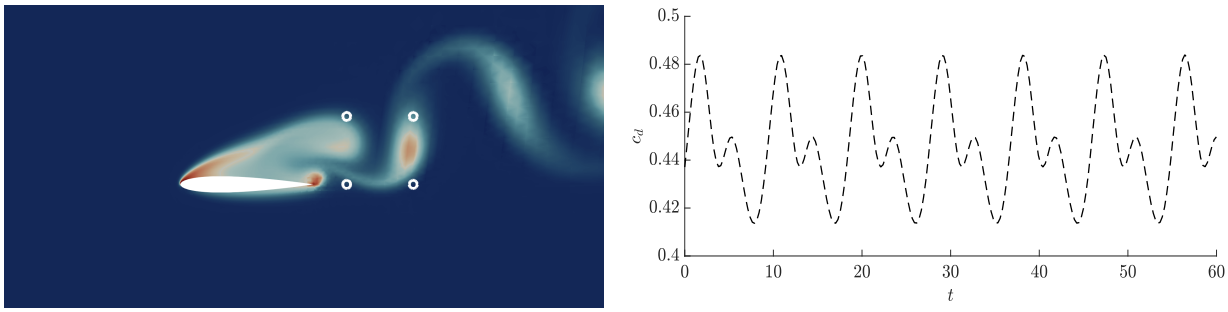
Remark 6. *The proposed framework contains several hyperparameters—the ensemble sizes (M_n and M_r), the RB and EQP tolerances (δ_{RB} and δ_{EQP}), the start time for the training data interval ($t^\star$), and the ROM retraining threshold (δ_u)—that are chosen manually. Some existing EnKF algorithms include estimates the optimal size of ensemble [33, 41], and adjust the size according. However, the automated and adaptive selection of optimal values for these hyperparameters is beyond the scope of this paper. We will numerically study the effect of varying the hyperparameters in Section 3.*

3. Numerical example

3.1. Problem description

We assess the effectiveness of the MFEnKF using a synthetic data assimilation problem associated with a separated flow past a NACA 0012 airfoil. We first introduce our "true" dynamics that govern the "true" state u^{true}. The flow is modeled by the (unsteady) compressible Navier-Stokes equations in entropy variables [6]. Namely, we seek the entropy variables

$$u \coloneqq \left(-\frac{s}{\gamma-1}+\frac{\gamma+1}{\gamma-1}-\frac{\rho e}{p},\frac{\rho v}{p},-\frac{\rho}{p}\right)$$



(a) "true" entropy field for $u^{\text{true},0}$

(b) "true" drag coefficient history

Figure 2: Dynamics of the model problem, showing entropy field at initial time for the "true" solution including velocity probe locations, and "true" drag coefficient as a function of time.

that satisfy the compressible Navier-Stokes equations

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho v) &= 0, \\
\frac{\partial \rho v}{\partial t} + \nabla \cdot (\rho v \otimes v + pI - \tau) &= 0, \\
\frac{\partial \rho e}{\partial t} + \nabla \cdot ((\rho e + p)v) - \tau v - \kappa \nabla T) &= 0,
\end{aligned}$$

where ρ is the density, v is the velocity, e is the specific internal energy, s is the thermodynamic entropy, p is the pressure, T is the temperature, τ is the stress tensor, and γ is the ratio of specific heats. We use a chord-based Reynolds number of $\text{Re}_c = 700$, a free stream Mach number of $M_\infty = 0.2$, and an angle of attack of $\alpha = 20°$. We consider a time interval of 60 non-dimesionalized time units; i.e., $I = (0, T = 60]$. A long-time integration of this problem yields a periodic response with a period of about 9.1 time units, and we take our initial condition $u^{\text{true},0}$ to be one randomly chosen state from this periodic response; i.e., $u^{\text{true},0}$ is randomly chosen from the set $U^{\text{init}} := \{u^{\text{true}}(t) \mid t > t^\odot \text{ for } t^\odot \text{ sufficiently large}\}$. Figure 2 shows a snapshot of the entropy field as well as the drag history of the solution.

We collect x- and y-velocity measurements at each point in a 2×2 grid located in the airfoil wake, as shown in Figure 2a. Each "point" probe is modeled as a Gaussian with the standard deviation of $0.1c$, where c is the chord of the airfoil, which is here taken to be 1. Note that the two-component measurements at 2×2 locations yield the observations $y^{\text{obs},k} \in \mathcal{Y} = \mathbb{R}^{n_o=8}$. We make observations at (non-dimesionalized) times of $\{12, 24, 36, 48\}$; i.e., $t^k = 12k$, $k = 1, \ldots, 4$. We assume that the observation noise is uncorrelated and the covariance is $\Gamma = 10^{-5} I$, where $I \in \mathbb{R}^{8\times 8}$ is the identity matrix.

To generate a "true" solution and the associated observations, we discretize the problem using an adaptive discontinuous Galerkin (DG) finite element method in space and the diagonally implicit Runge-Kutta (DIRK) method in time. Specifically, we discretize in space using an adaptively generated piecewise quadratic ($p = 2$) DG space with $n = 73{,}440$ degrees of freedom, which may be formally expressed as

$$\mathcal{U}_n := \{v \in L^2(D)^{m\equiv 4} \mid v|_\kappa \in \mathcal{P}^{p\equiv 2}(\kappa)^{m\equiv 4}, \forall \kappa \in \mathcal{T}_h\},$$

where m is the number of components in the state, D is the spatial domain, $\mathcal{P}^p(\kappa)$ is the space of polynomials of degree at most p over element κ, and $\mathcal{T}_h$ is a tesselation of D. We discretize in

time using a third-order DIRK method [1] with a time step of 0.25 (non-dimensional) time units. We solve the nonlinear problem using a parallel Newton-GMRES solver [39, 36]. In our synthetic example, this (computable) reference model provides the "true" dynamics (i.e., $\mathcal{M}^{\text{true}}$ and r^{true}) and observations (i.e., H^{true}, and $\{y^{\text{obs},k}\}_{k=1}^{5}$).

We take our FOM to be the same as the "true" dynamical model. In other words, $\mathcal{U}_n = \mathcal{U}$ is the aforementioned $n = 73{,}440$-dimensional DG approximation space, and our FOM operators are given by $\mathcal{M}_n = \mathcal{M}^{\text{true}}$, $r_n(\cdot) = r^{\text{true}}(\cdot)$, and $H_n(\cdot) = H^{\text{true}}(\cdot)$. Hence, the only imperfection in our FOM model comes from the unknown initial condition $u^{\text{true},0}$. Specifically, we assume that the initial condition is known to belong to the initial-condition set U^{init}, but we do not know which member of U^{init} is $u^{\text{true},0}$.

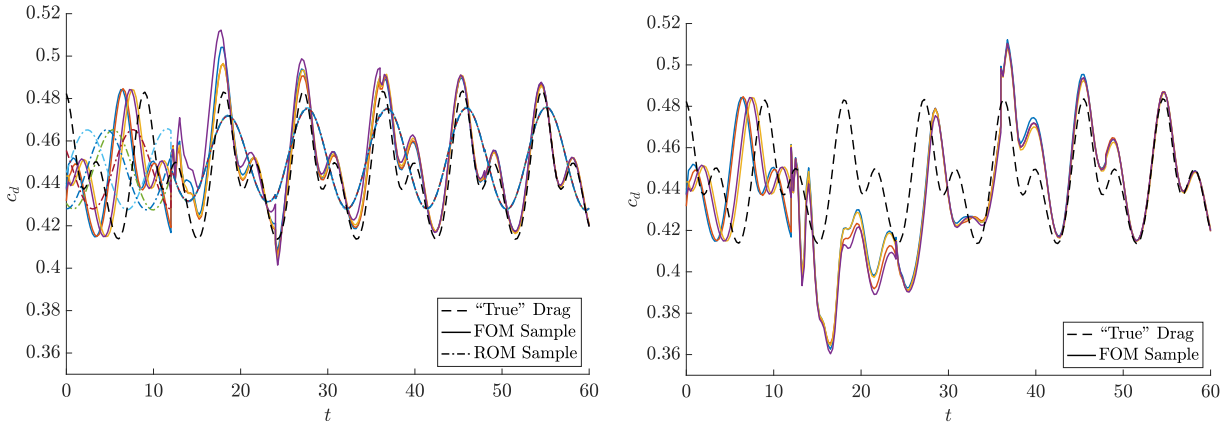
3.2. Illustration of the EnKF and the MFEnKF

We first illustrate the behavior of the (single-fidelity) EnKF and MFEnKF by comparing five different schemes:

(i) EnKF with $M_n = 4$.
(ii) EnKF with $M_n = 8$.
(iii) EnKF with $M_n = 64$. This filter simulates the EnKF in the mean-field ($M_n \to \infty$) limit.
(iv) MFEnKF with $M_n = 4$, $M_r = 128$, and $\delta_{\text{RB}} = \delta_{\text{RQ}} = 10^{-2}$. This filter uses a "lower-fidelity" ROM that is cheap enough to construct and evaluate so that the overall computational cost is comparable to that of case (i).
(v) MFEnKF with $M_n = 4$, $M_r = 128$, $\delta_{\text{RB}} = 10^{-4}$, and $\delta_{\text{RQ}} = 2 \times 10^{-3}$. This filter uses the same small FOM ensemble size as (i) but is augmented by an accurate "higher-fidelity" ROM ensemble.

For each method, we construct the initial ensemble by randomly drawing M_n or $M_n + M_r$ members, for EnKF and MFEnKF respectively, from the initial-condition set U^{init}. (Algorithm parameters are summarized in Table 1.) As the behavior of the EnKF and MFEnKF depend on the choice of the random initial ensemble and the observation noise, we consider nine different random seed initializations of each filter with different random seeds to assess the statistical behavior of the filters.

Before we discuss the results, we review the MFEnKF algorithm (Algorithm 1) as it applies to the present problem. The MFEnKF begins with $M_n = 4$ FOM evaluations over first forecast window $(t^0 = 0, t^1 = 12]$. Next, the $M_n = 4$ FOM solution trajectories are used to construct a hyperreduced ROM (i.e., the RB and RQ). We then invoke the ROM to rapidly propagate the ROM ensemble of the size $M_n + M_r = 4 + 128$ over the first forecast window $(t^0, t^1 = 12]$. Following this, we combine the FOM and ROM ensembles to compute the multi-fidelity Kalman gain at $t^1 = 12$ and incorporate the observation data $y^{\text{obs},1}$ to update the state estimates. The process repeats for time windows $[t^1, t^2]$, $[t^2, t^3]$, etc. However, in the subsequent time windows, the ROM reconstruction step is skipped if the ROM error metric (17) satisfies $\theta < \delta_u$, thereby reducing the total ROM training cost. For case (iv) we use $\delta_u = 0.90$ and $t^\star = t^{k-2}$ implying that the training set will contain data from at most the previous two assimilation windows; for case (v) we use we use $\delta_u = 0.55$ and $t^\star = t^{k-1}$ implying that the training set will contain data from only the directly previous assimilation window. The value of θ can vary with the random seed initialization, thereby varying the number of times the ROM is recomputed. For case (iv), it was most common to train the ROM after the first interval only, with this occurring for four of nine random seed initializations.



(a) Ensemble drag values with multi-fidelity estimates

(b) Ensemble drag values with single-fidelity estimates

Figure 3: The time history of the "true" drag and the drag estimated by members of the FOM and ROM ensembles in the MFEnKF with $M_n = 4$, $M_r = 128$, and $\delta_{\text{RB}} = \delta_{\text{RQ}} = 10^{-2}$, with and without the multi-fidelity contributions from the ROM ensemble, taken for the same random seed initialization.

For case (v), it was most common to train the ROM the first and second assimilation windows, with this being the case for four seeds.

We now qualitatively discuss the behavior of the MFEnKF. Figure 3a shows the time history of the drag estimates obtained by the MFEnKF scheme (iv) based on "lower-fidelity" ROMs. We observe that the drag estimate improves as we incorporate measurements at $t \in \{12, 24, 36, 48\}$ and the FOM ensemble members converge to the "true" drag history. Interestingly, the inexpensive ROM ($\delta_{\text{RB}} = \delta_{\text{RQ}} = 10^{-2}$) cannot capture the detailed dynamics of the flow and thus only poorly tracks the "true" drag; this however does not preclude the ROM from effectively reducing the error in the multi-fidelity estimate. The conditional ROM training retrains the (hyperreduced) ROM over only the first two time windows, $(0, 12]$ and $[12, 24]$; in other words, the set of eight trajectories associated with the $M_n = 4$ FOM ensemble over the two (observation) time intervals are sufficient to construct a ROM that is accurate over the entire time interval $(0, 60]$. This can be compared to the single-fidelity case, without the multi-fidelity estimate yielded by including the ROM, shown in Figure 3b; the two figures use the same random seed initialization, but the seemingly inaccurate ROM estimates greatly improve the quality of the updates and therefore the state estimate.

We now compare the five schemes (i)–(v) more quantitatively. To this end, we study the temporal variation of the L^2 spatial error. Namely, for the EnKFs ((i), (ii), and (iii)), we evaluate the L^2 error in the ensemble mean $\mathbb{E}^{\omega}_{M_n}[u_n(t)]$ with respect to the "true" state $u^{\text{true}}(t)$, $E(t) := \|u^{\text{true}}(t) - \mathbb{E}^{\omega}_{M_n}[u_n(t)]\|_{L^2(D)}$, where D is the spatial domain. For the MFEnKFs ((iv) and (v)), we evaluate the L^2 error in the multi-fidelity ensemble mean, $E(t) := \|u^{\text{true}}(t) - \mathbb{E}^{\omega,\tilde{\omega}}_{M_n,M_r}[u(t)]\|_{L^2(D)}$.

Figures 4a–4e show the time history of the L^2 error with nine separate random seed initializations for each of the five EnKF and MFEnKF schemes. Figure 4a shows that the EnKF with $M_n = 4$ (scheme (i)) is unstable and inaccurate, as the ensemble size is too small to compute an accurate Kalman gain. The EnKF scheme (ii) with $M_n = 8$ is shown in Figure 4b; it performs better than the $M_n = 4$ EnKF, but is twice as expensive, and still unreliable. Figure 4d shows results for the MFEnKF scheme (iv), which uses $M_n = 4$ along with $M_r = 128$ ROM samples trained with tolerances $\delta_{\text{RB}} = \delta_{\text{RQ}} = 10^{-2}$. Here, despite using the same number of FOM samples

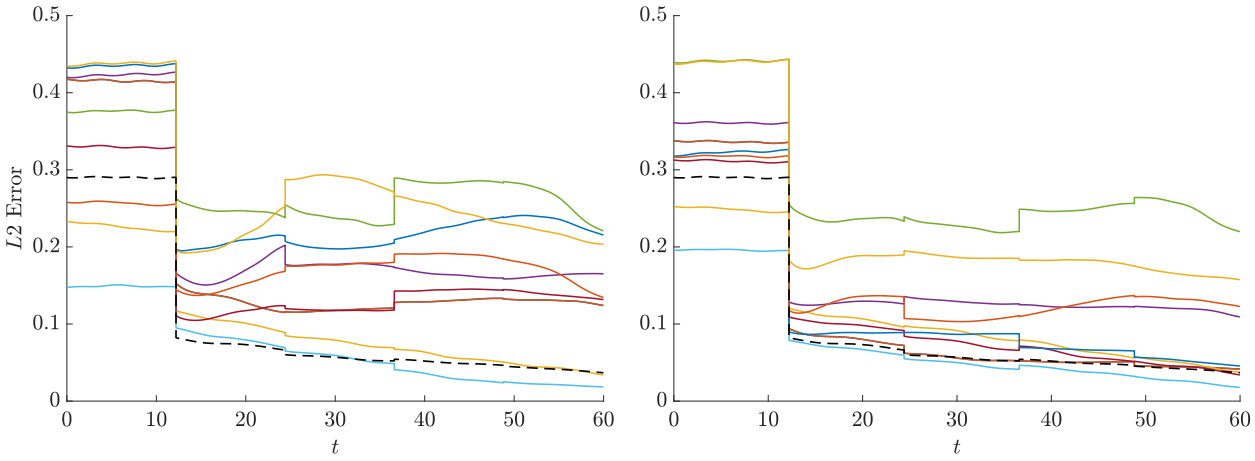
	algorithm parameters			mean wall-clock time		error $E(t=T)$	
	M_n	M_r	$\delta_{\text{RB}}, \delta_{\text{RQ}}$	ROM training	total	mean	mean+2std
(i)	4	-	-	-	$20t^{\text{fom}}$	0.137	0.276
(ii)	8	-	-	-	$40t^{\text{fom}}$	0.083	0.216
(iii)	64	-	-	-	$320t^{\text{fom}}$	0.037	0.059
(iv)	4	128	10^{-2}	$2t^{\text{fom}}$	$22t^{\text{fom}}$	0.057	0.096
(v)	4	128	$10^{-4}, 2\times10^{-3}$	$14t^{\text{fom}}$	$37t^{\text{fom}}$	0.043	0.060

Table 1: Summary of the computational cost and accuracy of the EnKF and MFEnKF schemes. The time unit t^{fom} is time required to march a single FE sample over one assimilation window.

as the EnKF scheme (i), the MFEnKF scheme (iv) obtains more accurate and reliable results; as we will see shortly, the construction and evaluation of the ROM increases the computational cost by only ≈10%. Similarly results for the MFEnKF scheme (v) are shown in Figure 4e, where the tolerances have been tightened to $\delta_{\text{RB}} = 10^{-4}$ and $\delta_{\text{RQ}} = 2\times10^{-3}$ as compared to scheme (iv). The MFEnKF scheme (v) not only yields more accurate and reliable results than the EnKF scheme (ii) but, as we will see shortly, also reduces the computational cost by ≈10%. Lastly, the "reference" EnKF scheme (iii) with $M_n = 64$ is shown in Figure 4c; it produces the most reliable and accurate state estimates, but, as we will see shortly, the computational costs are significantly higher than all other schemes. Figure 4f summarizes the results using the min-max envelopes for a few selected schemes.

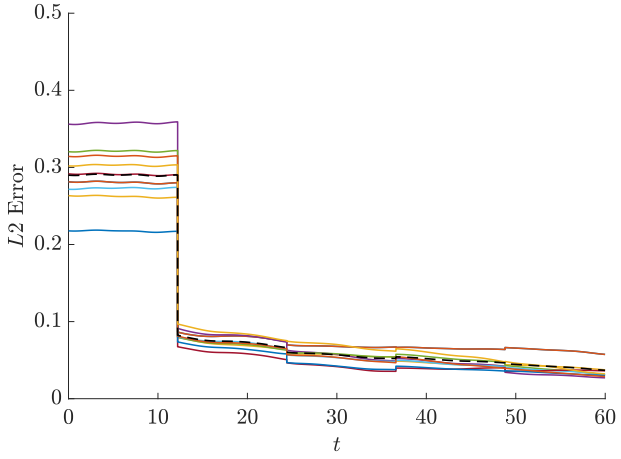
We now summarize the properties of the ROMs constructed by the MFEnKFs. For both MFEnKF cases (iv) and (v), each time window consists of 48 time steps, each with four DIRK stages, and $M_n = 4$ FOM ensembles, yielding a total of 772 temporal-ensemble training state snapshots after each time window. For scheme (v) with "higher-fidelity" ROMs, POD with $\delta_{\text{RB}} = 10^{-4}$ and $t^\star = t^{k-1}$ compresses the snapshots to yield a RB of the size $r \in [26, 57]$; this is a significant reduction from the FOM dimension of $n = 73{,}440$. (Since the sizes of the RB and RQ depend on the random initial ensembles and observation noises, we report the range observed over nine different random seed initializations.) Similarly, EQP with $\delta_{\text{RQ}} = 2\times10^{-3}$ reduces the number of quadrature points from $Q_n = 108{,}050$ for the FOM to $Q_r \in [361, 820]$. For scheme (iv) with "lower-fidelity" ROMs, with $\delta_{\text{RB}} = 10^{-2}$ and $t^\star = t^{k-2}$, the corresponding values are a RB of size $r = 3$ in all cases and a RQ of $Q_r \in [20, 27]$. Relative to the FOM, the computational time for a single ROM evaluation is reduced by $\approx 150\times$ and $\approx 350\times$ for the higher- and lower-fidelity ROMs, respectively.

Lastly, we summarize in Table 1 the algorithm parameters, the computational cost, and the error for the five cases in Figure 4. The table reports the total average wall-clock time—normalized by the time it takes to propagate FOM states over a single time window $[t^{k-1}, t^k]$—and the mean error with and without two added standard deviations, which represents some likely "worst case" for each scheme at the terminal time, T. We see that the cheapest option also produces the least reliable estimate and the most expensive option provides the best estimate, as expected. We note the main takeaway for this work: the MFEnKF can produce a state estimate that is more accurate and reliable than the standard EnKF, but at a cost that is comparable to taking only a few extra samples of a standard EnKF. We further note that the MFEnKF scheme (iv) outperforms the EnKF scheme (ii) in all metrics.

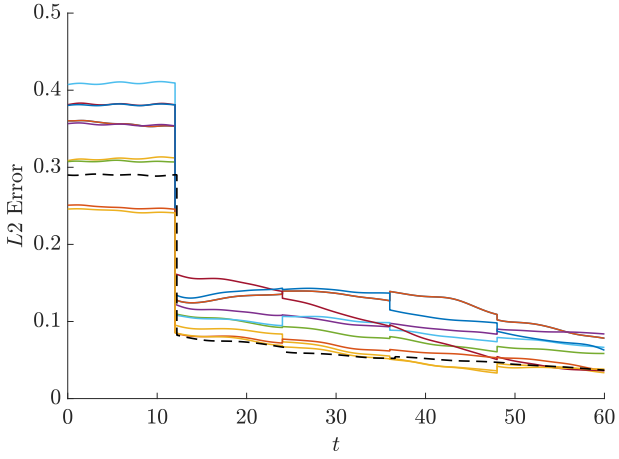


(a) EnKF: $M_n = 4$

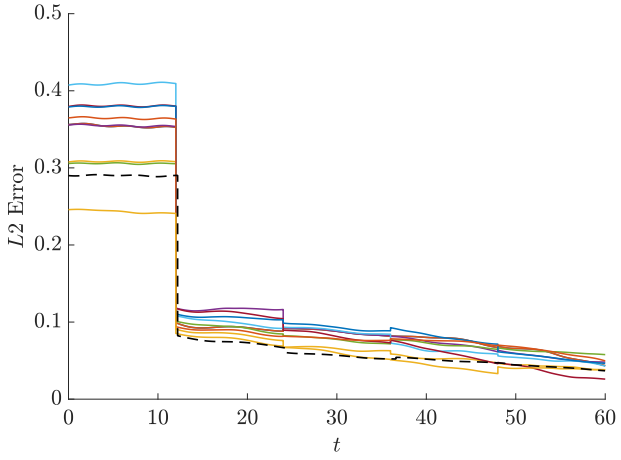
(b) EnKF: $M_n = 8$



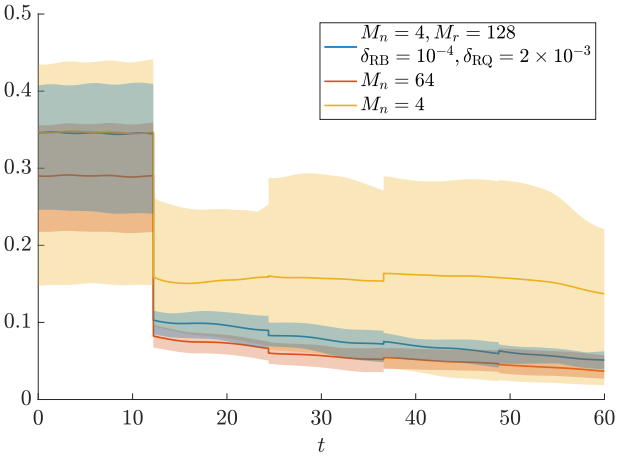
(c) EnKF: $M_n = 64$



(d) MFEnKF: $M_n = 4$, $M_r = 128$, $\delta_{\mathrm{RB}} = \delta_{\mathrm{RQ}} = 10^{-2}$

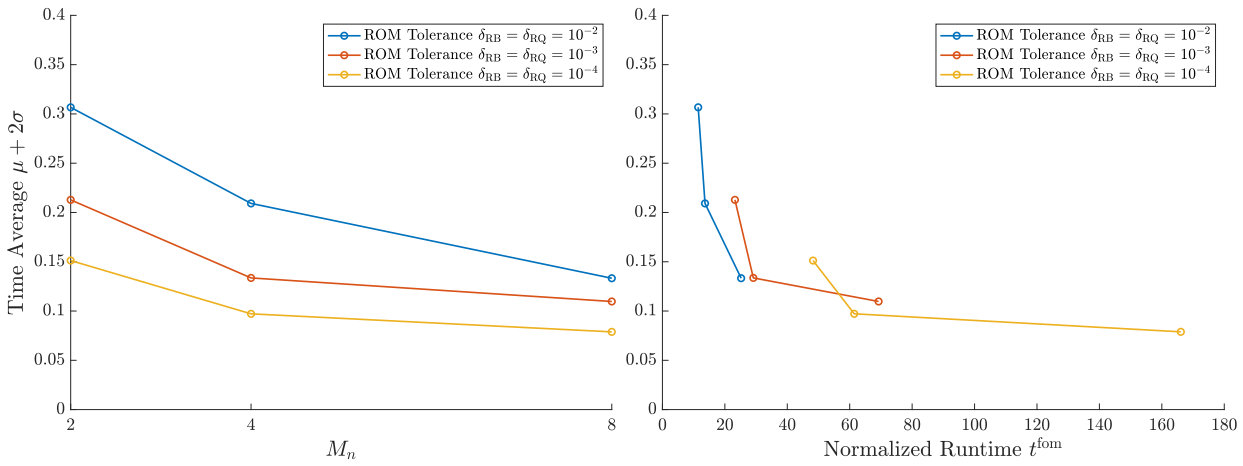


(e) MFEnKF: $M_n = 4$, $M_r = 128$, $\delta_{\mathrm{RB}} = 10^{-4}$, $\delta_{\mathrm{RQ}} = 2 \times 10^{-3}$



(f) Mean and min-max envelopes for selected schemes.

Figure 4: Time history of the L^2 error for EnKFs and MFEnKFs. Each solid line represents a single random seed initialization while the dashed line represents the mean-field limit.



(a) Error versus FOM ensemble size

(b) Error versus normalized runtime

Figure 5: MFEnKF error as a function of FOM ensemble size M_n and ROM tolerances (δ_{RB} and δ_{RQ})

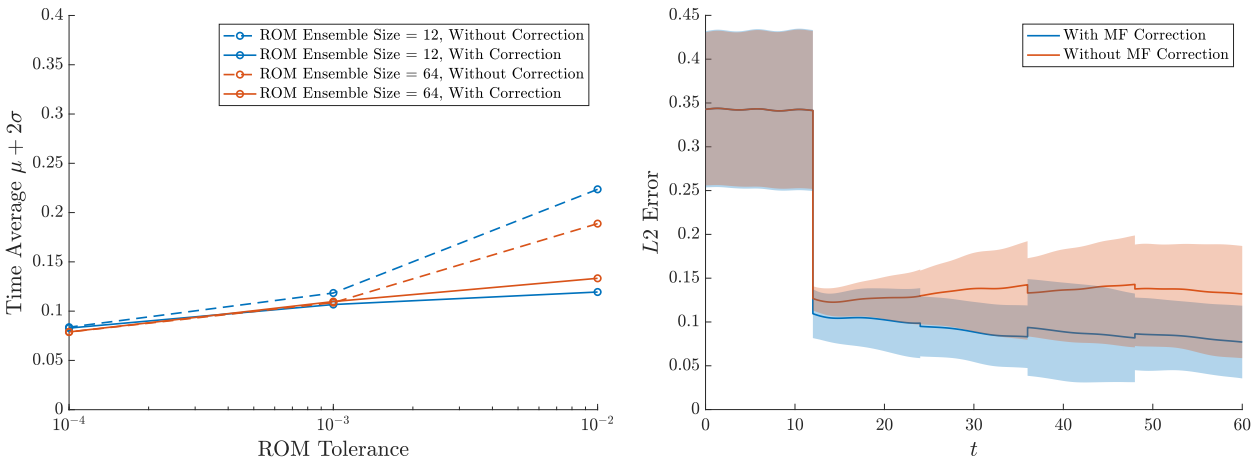
3.3. Effect of ROM fidelity and ROM ensemble size

We now study the effects of varying both the ROM tolerance $(\delta_{\text{RB}}, \delta_{\text{RQ}})$ and the FOM ensemble size M_n in the MFEnKF. In this study we fix the ROM ensemble size to $M_r = 64$; we can generally use a large ROM ensemble size at minimal costs thanks to the ≈150–350× speedup. The error metric used is the time averaged mean plus two standard deviations L^2 error—a probable worst-case scenario for each filter configuration—over the latter half of the entire time interval: i.e., $\text{mean}(\bar{E}) + 2\text{std}(\bar{E})$ for $\bar{E} \equiv \int_{t=30}^{60} E(t)dt$ with the mean and standard deviation over the nine random seed initializations. We exclude $t \in [0, 30]$ from the error metric because the errors strongly depend on the random initial ensemble rather than the filter performance.

Figure 5a shows the error metric decreases as either the ROM tolerance or FOM ensemble size increases, as expected. For a loose ROM tolerance, increasing the FOM ensemble size (which is used to train the ROM) more rapidly decreases the error as compared with a tight ROM tolerance. Since both tightening ROM tolerance and increasing FOM ensemble size can improve the filter performance, there exists a cost/effectiveness trade-off. If the ROM construction cost increases slowly with tighter ROM tolerance, then a small FOM ensemble with a tight ROM tolerance should be used; the converse holds if the ROM construction cost increases rapidly with tighter ROM tolerance. Figure 5b illustrates this trade-off by showing the error-versus-runtime Pareto front. For our aerodynamics problem, we observe that (a) $(M_n = 4, \delta_{\text{RB}} = \delta_{\text{RQ}} = 10^{-2})$ is most efficient at the 0.21 error level, (b) $(M_n = 8, \delta_{\text{RB}} = \delta_{\text{RQ}} = 10^{-2})$ and $(M_n = 4, \delta_{\text{RB}} = \delta_{\text{RQ}} = 10^{-3})$ are equally efficient at the ~ 0.13 error level, and (c) $(M_n = 4, \delta_{\text{RB}} = \delta_{\text{RQ}} = 10^{-4})$ is most efficient at the ~ 0.09 error level. In general, the accuracy benefit of increasing the FOM sample size and/or tightening the ROM tolerance saturates as we approach the mean-field limit for the filter.

3.4. Comparison of FOM-ROM MFEnKF and ROM-only EnKF

Given the effectiveness of the MFEnKF, we might ask about the role of the multi-fidelity correction from the difference of the principal and control variates compared to a single-fidelity ROM-based EnKF, which would be akin to using the ancillary variate only. To this end, we can compare (a) a single-fidelity ROM-based EnKF and (b) the MFEnKF. Both methods use



(a) $M_n = 8$, varying ROM tol. for $M_r = 12, 64$

(b) $M_n = 8, M_r = 64$, $\delta_{\text{RB}} = \delta_{\text{RQ}} = 10^{-2}$

Figure 6: Comparison of single-level EnKF based on a ROM (only) and MFEnKF based on both a FOM and ROM, showing effects of multi-fidelity correction.

$M_n = 8$ FOM ensemble to train the ROM. The ROM-based EnKF then uses $M_r = 12$ or $M_r = 64$ ROM ensemble to compute the (single-fidelity) Kalman gain in the EnKF; the MFEnKF uses $M_r + M_n = 12$ or $M_r + M_n = 64$ ROM ensemble and the $M_n = 8$ FOM ensemble to compute the multi-fidelity Kalman gain. We emphasize that the difference between these methods is whether or not the FOM is being used to correct the ROM statistical estimates themselves. Again, our goal is to assess the role of the multi-fidelity correction (for the same FOM and ROM ensemble sizes and hence computational cost).

Figure 6a shows the results. We can draw two important conclusions. First, with a tight ROM tolerance, the surrogate is accurate enough that the multi-fidelity correction has a negligible effect. As expected, a good enough ROM accurately approximates the FOM and the correction term is small. Secondly, when the ROM is inaccurate the multi-fidelity correction can reduce the error by a factor of approximately 1/2. We emphasize that (i) the cost of this multi-fidelity correction is negligible, as the FOM states must be computed anyway to train the ROM; and (ii) using a single-fidelity ROM-based EnKF requires an accurate ROM, but using a MFEnKF is effective even when the construction of an accurate ROM is prohibitive due to complex dynamics or unfavorable due to high training cost.

4. Summary and perspectives

This work develops and assesses a MFEnKF, a data assimilation method for nonlinear dynamical systems which uses multi-fidelity statistical estimates based on a FOM and hyperreduced ROMs. The MFEnKF effectively reduces the cost of state estimation for an aerodynamics problem by exploiting both hyperreduced ROMs constructed on-the-fly along with multi-fidelity Monte Carlo methods to achieve more rapid convergence than the standard single-fidelity EnKF. The method however has some limitations: several hyperparameters must be tuned for each problem; and constructing an effective ROM for more complicated problems is challenging, e.g. turbulent flows. Future work for this framework include the development of an adaptive method which automatically

select each of the aforementioned hyperparameters and exploring nonlinear model reduction schemes to deal with more complicated flow features.

Acknowledgment

The financial support for this work was provided by the Connaught Fund and the Natural Sciences and Engineering Research Council of Canada. Computations were performed on the Niagara supercomputer at the SciNet HPC Consortium. SciNet is funded by the Canada Foundation for Innovation; the Government of Ontario; Ontario Research Fund - Research Excellence; and the University of Toronto. We would also like to thank anonymous reviewers for very helpful feedback that has improved this manuscript.

References

[1] R. Alexander. Diagonally implicit Runge-Kutta methods for stiff O.D.E.'s. *SIAM Journal on Numerical Analysis*, 14(6):1006–1021, 1977.

[2] J. L. Anderson. An ensemble adjustment Kalman filter for data assimilation. *Monthly Weather Review*, 129(12):2884–2903, Dec 2001.

[3] J. L. Anderson. An adaptive covariance inflation error correction algorithm for ensemble filters. *Tellus A*, 59(2):210–224, 2007.

[4] N. Aretz-Nellesen, M. A. Grepl, and K. Veroy. 3D-VAR for parameterized partial differential equations: a certified reduced basis approach. *Advances in Computational Mathematics*, 45(5-6):2369–2400, Jul 2019.

[5] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An "empirical interpolation" method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Acad. Sci. Paris, Ser. I*, 339:667—672, 2004.

[6] T. J. Barth. Numerical methods for gasdynamic systems on unstructured meshes. In D. Kröner, M. Ohlberger, and C. Rohde, editors, *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws*, pages 195–282. Springer-Verlag, 1999.

[7] P. Benner, S. Gugercin, and K. Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531, 2015.

[8] K. Carlberg, C. Bou-Mosleh, and C. Farhat. Efficient non-linear model reduction via a least-squares Petrov-Galerkin projection and compressive tensor approximations. *International Journal for Numerical Methods in Engineering*, 86(2):155–181, 2011.

[9] S. Chaturantabut and D. C. Sorensen. Nonlinear model reduction via Discrete Empirical Interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

[10] A. Chernov, H. Hoel, K. J. H. Law, F. Nobile, and R. Tempone. Multilevel ensemble Kalman filtering for spatio-temporal processes. *Numerische Mathematik*, Nov 2020.

[11] A. F. da Silva and T. Colonius. Ensemble-based state estimator for aerodynamic flows. *AIAA Journal*, 56(7):2568–2578, 2018.

[12] M. Dihlmann and B. Haasdonk. A reduced basis Kalman filter for parametrized partial differential equations. *ESAIM: Control, Optimisation and Calculus of Variations*, 22(3):625–669, Apr 2016.

[13] G. Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5):10143–10162, 1994.

[14] G. Evensen. *Data assimilation: the ensemble Kalman filter*. Springer, 2009.

[15] C. Farhat, P. Avery, T. Chapman, and J. Cortial. Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency. *International Journal for Numerical Methods in Engineering*, 98(9):625–662, 2014.

[16] K. J. Fidkowski and P. L. Roe. An entropy adjoint approach to mesh refinement. *SIAM Journal on Scientific Computing*, 32(3):1261–1287, Jan 2010.

[17] M. B. Giles. Multilevel Monte Carlo path simulation. *Operations Research*, 56(3):607–617, 2008.

[18] M. B. Giles. Multilevel Monte Carlo methods. *Acta Numerica*, 24:259–328, 2015.

[19] P. W. Glynn and R. Szechtman. Some new perspectives on the method of control variates. pages 27–49, 2002.

[20] A. Gregory, C. J. Cotter, and S. Reich. Multilevel ensemble transform particle filtering. *SIAM Journal on Scientific Computing*, 38(3):A1317–A1338, Jan 2016.

[21] S. J. Greybush, E. Kalnay, T. Miyoshi, K. Ide, and B. R. Hunt. Balance and ensemble Kalman filter localization techniques. *Monthly Weather Review*, 139(2):511 – 522, Feb. 2011.

[22] A. Harten. On the symmetric form of systems of conservation laws with entropy. *Journal of Computational Physics*, 49(1):151–164, Jan 1983.

[23] J. He, P. Sarma, and L. J. Durlofsky. Use of reduced-order models for improved data assimilation within an EnKF context. In *SPE Reservoir Simulation Symposium*. Society of Petroleum Engineers, 2011.

[24] H. Hoel, K. J. H. Law, and R. Tempone. Multilevel ensemble Kalman filtering. *SIAM Journal on Numerical Analysis*, 54(3):1813–1839, Jan 2016.

[25] H. Hoel, G. Shaimerdenova, and R. Tempone. Multilevel ensemble Kalman filtering based on a sample average of independent EnKF estimators. *Foundations of Data Science*, 2(4):351, 2020.

[26] P. L. Houtekamer and F. Zhang. Review of the ensemble Kalman filter for atmospheric data assimilation. *Monthly Weather Review*, 144(12):4489–4532, Dec 2016.

[27] T. J. R. Hughes, L. P. Franca, and M. Mallet. A new finite element formulation for computational fluid dynamics: I Symmetric forms of the compressible Euler and Navier-Stokes equations and the second law of thermodynamics. *Comput. Methods Appl. Mech. Engrg.*, 54:223–234, 1986.

[28] A. Jasra, K. Kamatani, K. J. Law, and Y. Zhou. Multilevel particle filters. *SIAM Journal on Numerical Analysis*, 55(6):3068–3096, 2017.

[29] M. Kaercher, S. Boyaval, M. A. Grepl, and K. Veroy. Reduced basis approximation and a posteriori error bounds for 4D-Var data assimilation. *Optimization and Engineering*, 19(3):663–695, 2018.

[30] R. Kikuchi, T. Misaka, and S. Obayashi. Assessment of probability density function based on POD reduced-order model for ensemble-based data assimilation. 47(5):051403, Sep 2015.

[31] K. Law, A. Stuart, and K. Zygalakis. *Data Assimilation*. Springer International Publishing, 2015.

[32] B. Lin and D. McLaughlin. Efficient characterization of uncertain model parameters with a reduced-order ensemble Kalman filter. *SIAM Journal on Scientific Computing*, 36(2):B198–B224, Jan 2014.

[33] H. L. Mitchell and P. L. Houtekamer. An adaptive ensemble Kalman filter. *Monthly Weather Review*, 128(2):416, 2000.

[34] P. Mycek and M. D. Lozzo. Multilevel Monte Carlo covariance estimation for the computation of Sobol' indices. *SIAM/ASA Journal on Uncertainty Quantification*, 7(4):1323–1348, jan 2019.

[35] S. Pagani, A. Manzoni, and A. Quarteroni. Efficient state/parameter estimation in nonlinear unsteady PDEs by a reduced basis ensemble Kalman filter. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):890–921, Jan 2017.

[36] P.-O. Persson and J. Peraire. Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier-Stokes equations. *SIAM Journal on Scientific Computing*, 30(6):2709–2733, Jan 2008.

[37] A. A. Popov, C. Mou, A. Sandu, and T. Iliescu. A multifidelity ensemble Kalman filter with reduced order control variates. *SIAM Journal on Scientific Computing*, 43(2):A1134–A1162, Jan 2021.

[38] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations — Application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.

[39] Y. Saad and M. H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869, 1986.

[40] M. K. Sleeman and M. Yano. Goal-oriented model reduction for parametrized time-dependent nonlinear partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 388:114206, Jan 2022.

[41] B. Uzunoglu, S. J. Fletcher, M. Zupanski, and I. M. Navon. Adaptive ensemble reduction and inflation. *Quarterly Journal of the Royal Meteorological Society*, 133(626):1281–1294, 2007.

[42] F. Vidal-Codina, N. C. Nguyen, M. B. Giles, and J. Peraire. A model and variance reduction method for computing statistical outputs of stochastic elliptic partial differential equations. *Journal of Computational Physics*, 297:700–720, 2015.

[43] D. Xiao, J. Du, F. Fang, C. Pain, and J. Li. Parameterised non-intrusive reduced order methods for ensemble Kalman filter data assimilation. *Computers & Fluids*, 177:69–77, Nov 2018.

[44] M. Yano. Goal-oriented model reduction of parametrized nonlinear PDEs; application to aerodynamics. *International Journal for Numerical Methods in Engineering*, 121(23):5200–5226, 2020.

[45] M. Yano and A. T. Patera. An LP empirical quadrature procedure for reduced basis treatment of parametrized nonlinear PDEs. *Computer Methods in Applied Mechanics and Engineering*, 344:1104–1123, Feb 2019.

[46] Y. Yue and K. Meerbergen. Accelerating optimization of parametric linear systems by model order reduction. *Society for Industrial and Applied Mathemtaics Journal of Optimization*, 23(2):1344–1370, 2013.

[47] M. J. Zahr and C. Farhat. Progressive construction of a parametric reduced-order model for PDE-constrained optimization. *International Journal for Numerical Methods in Engineering*, 102(5):1111–1135, Dec 2015.