Training physics-based machine-learning parameterizations with gradient-free ensemble Kalman methods

Ignacio Lopez-Gomez[1], Costa Christopoulos[1], Haakon Ludvig Langeland Ervik[1], Oliver R. A. Dunbar[1], Yair Cohen[1], Tapio Schneider[1,2]

[1]Department of Environmental Science and Engineering, California Institute of Technology, Pasadena, CA, USA.
[2]Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, USA.

Key Points:

- Ensemble Kalman methods can be used to train parameterizations regardless of their architecture.
- They enable learning from partial observations or statistics in the presence of noise.
- Their effectiveness is demonstrated by calibrating an atmospheric turbulence and convection model.

Corresponding author: Ignacio Lopez-Gomez, ilopezgo@caltech.edu

Abstract

Most machine learning applications in Earth system modeling currently rely on gradient-based supervised learning. This imposes stringent constraints on the nature of the data used for training (typically, residual time tendencies are needed), and it complicates learning about the interactions between machine-learned parameterizations and other components of an Earth system model. Approaching learning about process-based parameterizations as an inverse problem resolves many of these issues, since it allows parameterizations to be trained with partial observations or statistics that directly relate to quantities of interest in long-term climate projections. Here we demonstrate the effectiveness of Kalman inversion methods in treating learning about parameterizations as an inverse problem. We consider two different algorithms: unscented and ensemble Kalman inversion. Both methods involve highly parallelizable forward model evaluations, converge exponentially fast, and do not require gradient computations. In addition, unscented Kalman inversion provides a measure of parameter uncertainty. How learning about parameterizations can be posed as an inverse problem and solved by ensemble Kalman methods is illustrated through the calibration of an eddy-diffusivity mass-flux scheme for subgrid-scale turbulence and convection, using data generated by large-eddy simulations. We find the algorithms amenable to batching strategies, robust to noise and model failures, and efficient in the calibration of hybrid parameterizations that can include empirical closures and neural networks.

Plain Language Summary

Artificial intelligence represents an exciting opportunity in Earth system modeling, but its application brings its own set of challenges. One of these challenges is to train machine learning systems within Earth system models from partial data. Here we present algorithms, known as ensemble Kalman methods, that can be used to train such systems. We demonstrate their use in situations where the data used for training are noisy, only indirectly informative about the model to be trained, and may only become available sequentially. As an example, we present training results for a state-of-the-art model for turbulence, convection, and clouds for use within Earth system models. This model is shown to learn efficiently from data in a variety of configurations, including situations where the model contains neural networks.

1 Introduction

The remarkable achievements of machine learning over the past decade have led to renewed interest in informing Earth system models with data (Schneider et al., 2017; Reichstein et al., 2019). The spotlight is often on creating or improving models of processes that are deemed important for the correct representation of the Earth system as a whole. Examples of these processes include moist convection (Brenowitz et al., 2020), cloud microphysical and radiative effects (Seifert & Rasp, 2020; Villefranque et al., 2021; Meyer et al., 2022), and evapotranspiration (Zhao et al., 2019), among others.

Processes governed by poorly understood dynamics, such as cloud microphysics, are obvious candidates for representation by purely data-driven models. On the other end of the spectrum are fluid transport processes, which are governed by the Navier-Stokes equations. Uncertain representation of these processes comes from a lack of resolution, not lack of knowledge about the underlying dynamics. Hybrid modeling approaches that incorporate domain knowledge and augment it by learning from data are attractive for such processes, because they reduce what needs to be learned from data.

For processes with known dynamics, data-informed models fall into three broad categories according to their leverage of domain knowledge. In the first category are models that try to learn the entire dynamics using a sufficiently expressive hypothesis set,

such as deep neural networks. This approach has proved successful for predicting precipitation over short time horizons (Ravuri et al., 2021), and it has been explored for medium-range weather forecasting (Rasp & Thuerey, 2021; Pathak et al., 2022). An advantage of these models is that they are typically easy to implement and cheap to evaluate. They can afford very large time steps (Weyn et al., 2021), or they may learn directly mappings from the initial state to a probability distribution of final states with no need of time marching or ensemble forecasting (Sønderby et al., 2020). A deficiency of these models is that they often require an extreme amount of data to constrain the many (often $> 10^6$) parameters in them and achieve acceptable performance.

Methods in the second and third categories employ models of subgrid processes to solve the closure problem that arises when coarse-graining the known dynamics, which are retained. Retaining the coarse-grained equations of motion ensures conservation of mass, momentum, and energy, which is more difficult when using models in the first category (Beucler et al., 2021; Brenowitz et al., 2020). The second category encompasses methods that try to learn the functional form of these closures avoiding the use of empirical laws. For example, Zanna and Bolton (2020) use relevance vector machines to prune a library of functions and find a closed form expression of mesoscale eddy fluxes in ocean simulations; Ling et al. (2016) learn a neural network closure of the Reynolds stress anisotropy tensor while explicitly encoding rotational invariance, in the context of $k-\epsilon$ models of turbulence.

Finally, the third category refers to methods that seek to learn the parameters that arise in empirical closures of subgrid processes. In general, models in the third category are more restrictive, and they may be expected to underperform with respect to those in the second category given sufficient data on the target distributions. However, the limited parametric complexity of these closures makes them amenable to physical interpretation, robust to overfitting, and better suited for learning in the low-data regime. This may be attractive for Earth system models, for which online learning from limited high-resolution data may be a useful strategy to assimilate computationally generated data of the changing climate (Schneider et al., 2017).

A barrier delimiting data-driven and empirical subgrid-scale closures is the access to practical calibration tools. Neural network parameterizations are easily calibrated using stochastic gradient descent through backpropagation, which limits datasets to those including output labels, and models to those that afford automatic differentiation with respect to their parameters. Empirical closures, which may depend on time-evolving terms with memory (e.g., Lopez-Gomez et al., 2020) or yield unobservable outputs (e.g., turbulent versus dynamical entrainment in Cohen et al., 2020) cannot be trained using the same approach. Techniques developed to train empirical models are often computationally expensive and may scale poorly with the number of parameters (Couvreux et al., 2021), which can limit their application to data-driven closures with many parameters. Model-agnostic tools that enable fast calibration of subgrid-scale closures from diverse data are a necessary step toward the development of hybrid closures that leverage the strengths of all modeling approaches.

With this goal in mind, we present calibration strategies for models of subgrid processes, formulating the learning task as an inverse problem (Kovachki & Stuart, 2019). Solutions to the inverse problem are sought using the ensemble and unscented Kalman inversion algorithms (Iglesias et al., 2013; Huang et al., 2022). Emphasis is given to practical aspects of this specific inverse problem, which have not previously been explored in the literature. These include the construction of a domain-agnostic loss function from high-dimensional observations, a heuristic a priori estimate of model error, systematic handling of model failures, and the use of the Kalman inversion algorithms when only noisy evaluations of the loss function are available.

The strategies presented here are designed to have several attractive properties compared to other learning algorithms. First, framing learning as an inverse problem enables the use of partial observations or statistical summaries of the data. Second, calibration is performed using gradient-free methods, well suited for stochastic models and/or models whose derivatives do not exist or are difficult to obtain. Finally, the strategies presented are amenable to massive parallelization and the use of high-dimensional correlated observations. The last two properties draw heavily on the use of the recently developed family of Kalman inversion algorithms to tackle the inverse problem. The methods presented are applicable to models of subgrid-scale processes, within the second and third categories described above. They provide an alternative to learning algorithms that impose stringent requirements on either the model architecture or the nature of the training data.

The article is organized as follows. Section 2 casts learning about parameterizations as an inverse problem, which can be solved through the minimization of a low-dimensional encoding of the data-model mismatch. Section 3 reviews the application of the ensemble and unscented Kalman inversion algorithms to inverse problems. Section 4 then applies these ensemble Kalman algorithms to the calibration of closures within an eddy-diffusivity mass-flux (EDMF) scheme of turbulence and convection, using data generated from large-eddy simulations (LES). The robustness of these learning strategies is demonstrated by calibrating the EDMF scheme using noisy loss evaluations and partial information, and their flexibility is emphasized by learning the parameters in a hybrid model containing both empirical and neural network closures. Finally, Section 5 ends with a discussion of the findings and some concluding remarks.

2 Learning about parameterizations as an inverse problem

We consider the problem of learning the parameters ϕ of a dynamical model $\Psi(\phi)$, using noisy observations y of the true dynamical system ζ that $\Psi(\phi)$ seeks to represent. In the context of subgrid parameterizations, $\Psi(\phi)$ represents a closed version of the coarse-grained dynamical system (e.g., the filtered Navier-Stokes equations), where closures are parameterized by ϕ. The model $\Psi(\phi)$ maps a user-defined initial state φ_0 and a forcing $F_\varphi(t)$ to a state trajectory $\tilde{\varphi}(t)$. Thus, our definition of $\Psi(\phi)$ can be interpreted as the iterative application of the resolvent operator on the initial field φ_0 (Brajard et al., 2021). In the following, we denote any set of initial and forcing conditions collectively as the configuration $x_c = \{\varphi_0, F_\varphi\}_c$.

For each configuration x_c, the dynamical model can be related to the observations y_c by the observational map $\mathcal{H}_c$, which encapsulates all averaging and post-processing operations necessary to yield the model predictions associated with the observations. More precisely, the relationship between the dynamical model, the true dynamics, and the observations for a given configuration may be expressed as

$$y_c = \mathcal{H}_c \circ \zeta(x_c) + \eta_c = \mathcal{H}_c \circ \Psi(\phi; x_c) + \delta(x_c) + \eta_c, \tag{1}$$

where y_c are the observations associated with x_c, ζ is the true dynamical system, $\phi \in \mathbb{R}^p$ is the vector of learnable parameters, η_c is the observational noise associated with y_c and $\delta(\cdot)$ is the model error, which is a function of the configuration (Kennedy & O'Hagan, 2001).

Observations are taken to come from finite spatial and temporal averages of fields such as temperature. Learning from averages can help prevent overfitting to trajectories in chaotic systems by focusing on the statistics of the dynamics (Morzfeld et al., 2018) and improve numerical stability when coupling to a parent model (Brenowitz & Bretherton, 2018). Under this definition of observations, it is reasonable to assume the noise η_c to be additive and Gaussian, based on the central limit theorem (Cleary et al., 2021). In the following, we will further consider $\delta(\cdot)$ to be a centered Gaussian, although this

constitutes a significantly stronger assumption (e.g., that the model is unbiased) and may not be appropriate for the characterization of posterior uncertainty. These assumptions enable us to write $\delta(x_c) + \eta_c \sim \mathcal{N}(0, \Gamma_c)$.

In general, we are interested in minimizing the mismatch between y_c and the model output for a wide range of configurations $C = \{x_c,\ c = 1, \ldots, |C|\}$, representative of the conditions in which the model will operate. Therefore, the task of learning a set of model parameters ϕ can be cast as the inverse problem

$$y = \mathcal{H} \circ \Psi(\phi) + \delta + \eta, \tag{2}$$

where $y = [y_1, \ldots, y_{|C|}]^T \in \mathbb{R}^d$, $\delta = [\delta(x_1), \ldots, \delta(x_{|C|})]^T$, $\eta = [\eta_1, \ldots, \eta_{|C|}]^T$, $\mathcal{H} \circ \Psi(\phi) = [\mathcal{H}_1 \circ \Psi(\phi; x_1), \ldots, \mathcal{H}_{|C|} \circ \Psi(\phi; x_{|C|})]^T$ and $\delta + \eta \sim \mathcal{N}(0, \Gamma)$. In addition, implicit in the definition of the dynamical model $\Psi(\phi)$ is a discrete resolution Δ. This dependence may be lifted if the closures are designed to be scale-aware or scale-independent, in which case the inverse problem (2) should be augmented by stacking copies of y and evaluating $\mathcal{H} \circ \Psi(\phi, \Delta_i)$ for different discretizations Δ_i.

In practice, the parameters ϕ are often defined over some subspace $U \subset \mathbb{R}^p$, outside of which the trajectories given by $\Psi(\phi)$ are either unphysical or dominated by numerical instabilities. Examples of these are parameters controlling the intensity of diffusion or turbulent dissipation of a scalar field, for which negative values are not physically valid. On the other hand, many algorithms designed to solve inverse problems of the form (2) assume $\phi \in \mathbb{R}^p$. This obstacle may be circumvented by defining a transformation $\mathcal{T} : U \to \mathbb{R}^p$ (Dunbar et al., 2022), such that the inverse problem can be defined in an unconstrained parameter space,

$$y = \mathcal{G}(\theta) + \delta + \eta, \tag{3}$$

where

$$\mathcal{G} \equiv \mathcal{H} \circ \Psi \circ \mathcal{T}^{-1}, \qquad \phi = \mathcal{T}^{-1}(\theta). \tag{4}$$

In expressions (3) and (4), $\theta \in \mathbb{R}^p$ is the parameter vector in unconstrained space and $\mathcal{G} : \mathbb{R}^p \to \mathbb{R}^d$ is the map from transformed parameters to model predictions, which in the context of the inverse problem (3) represents the forward model. Note that the observational map $\mathcal{H}_c$ and the error covariance Γ_c defining the model-data relation (1) are yet to be defined. In the following subsections, we suggest definitions of these terms relevant to the calibration of models with an unknown error structure $\delta(\cdot)$.

2.1 Application to problems with high-resolution observations

High-resolution data are becoming increasingly common, from PDE solvers such as LES (Pressel et al., 2015; Shen et al., 2022), reanalysis products (Muñoz-Sabater et al., 2021), and satellite imagery (Schmit et al., 2017). Although computationally generated and thus suffering from their own limitations (e.g., microphysical processes still need to be parameterized even in LES), data from PDE solvers have some particularly desirable properties for the calibration of dynamical models:

- All prognostic variables and tendencies appearing in the coarse-grained equations of motion are observable. As a consequence, the nature of the observational map $\mathcal{H}$ used to constrain the model is largely a design choice.
- Data can be obtained systematically for all configurations x_c of interest, which may be optimized to minimize parameter uncertainty (Dunbar et al., 2022). In contrast, data drawn from physical experiments or field measurements are often sparse in the space of forcing and boundary conditions.

High-resolution data are often high-dimensional, which poses particular difficulties regarding the conditioning and tractability of linear systems of equations when solving in-

verse problem (3). The guidelines presented in this section are tailored to solve these issues, with a focus on synthetic data from high-fidelity solvers.

2.1.1 Estimate of noise covariances

The use of synthetic high-resolution data has implications for the noise structure of the inverse problem (3). Due to the tight coupling between resolution and accuracy of computational solvers, the observational noise on averaged quantities is typically small compared to the model error, so the leading order error in (3) comes from δ. However, since the structure of δ is unknown a priori, we must either parameterize it and calibrate it as well (Kennedy & O'Hagan, 2001), or use a heuristic to capture its magnitude. Here, we follow the second route and offer a heuristic that has worked well for us in practice.

If we consider the uncertainty in x_c to be negligible, and take $\mathcal{H}_c$ to be a measurement of the state aggregated over a time interval τ, we can write (1) as

$$\varphi_{\text{obs}}(t) - \varphi_0 = \tilde{\varphi}(t) - \varphi_0 + \delta(x_c) + \eta_c, \tag{5}$$

where $\varphi_{\text{obs}}(t)$ and $\tilde{\varphi}(t)$ are the observed and predicted measures centered at time t, respectively. If we consider a model with no predictive power such that $\tilde{\varphi}(t) \approx \varphi_0$ for all times t, and take the covariance of (5) from $t = 0$ to $t = T_\Gamma \gg \tau$,

$$\Gamma_c = \text{Var}(\varphi_{\text{obs}}) \approx \text{Var}(\delta(x_c)) + \text{Var}(\eta_c), \tag{6}$$

The aggregate noise $\eta_c + \delta(x_c) \sim \mathcal{N}(0, \Gamma_c)$ is estimated from the variability of the observed field φ_{obs} over a time interval T_Γ from known initial conditions φ_0. Note that for non-stationary conditions or finite-time averages, Γ_c depends on T_Γ. We emphasize that the heuristic (6) is most appropriate when observations y_c are obtained from a synthetic system ζ that accepts the same configuration x_c as model $\Psi(\phi)$, as is the case when ζ is a high-fidelity PDE solver.

2.2 Design of the observational map

2.2.1 Metric-based calibration and model calibration

The observations y in (3) can be chosen to represent a summary of the data obtained through some engineered transformation $\mathcal{H}$ whose definition involves domain-specific knowledge (Couvreux et al., 2021). This is natural when trying to optimize a particular metric, like cloud cover, for which we denote this approach *metric-based calibration*. In contrast, we define *model calibration* as the minimization of the mismatch between the observed coarse-grained dynamics and the dynamics induced by the model. We will use this definition to construct a domain-agnostic map $\mathcal{H}$. As an example, consider a system ζ with coarse-grained dynamics

$$\frac{\partial \bar{\varphi}}{\partial t} + \bar{\boldsymbol{v}} \cdot \nabla \bar{\varphi} + \nabla \cdot (\overline{\boldsymbol{v}'\varphi'}) = F_\varphi, \tag{7}$$

where $\bar{(\cdot)}$ denotes spatial filtering, $(\cdot)'$ subfilter-scale fluctuations, and F_φ is the forcing. The field $\bar{\boldsymbol{v}}$ is prescribed and $\overline{\boldsymbol{v}'\varphi'}$ is the term parameterized in $\Psi(\phi)$. Let $S(t) = [\bar{\varphi}(t),\ \overline{\boldsymbol{v}'\varphi'}(t)]^T$ be the true closed state, and $\tilde{S}(t)$ the closed state predicted by the model. For a compressible fluid model, $S(t)$ would contain the fluid density, momentum, energy and the subgrid advective fluxes of these fields.

We define model calibration as finding the minimizer of the expected data mismatch $\mathbb{E}[\tilde{S} - S]$ with respect to some norm and time interval, for known initial conditions and forcing F_φ. We minimize the expected mismatch to allow for the calibration of stochastic models. Observations of the closed state $S(t)$ are not always available, and so this definition of model calibration is representative of the ideal learning scenario. In any other scenario, we will consider $S(t)$ to be formed by all relevant observable spatial fields.

2.2.2 Observations in physical space

Following our definition of model calibration, we preliminarily define the observations in the model-data relation (1) as finite-time averages of the normalized observed state s_c for a set of configurations C,

$$\tilde{y}_c = \frac{1}{T_c}\int_{t_c-T_c}^{t_c} s_c(\tau)d\tau, \qquad s_c = \begin{bmatrix} \tilde{v}_{c,1} \\ \dots \\ \tilde{v}_{c,n_c} \end{bmatrix} = \begin{bmatrix} \sigma_{c,1}^{-1}\tilde{V}_{c,1} \\ \dots \\ \sigma_{c,n_c}^{-1}\tilde{V}_{c,n_c} \end{bmatrix}, \qquad c = 1, \dots |C|, \tag{8}$$

where T_c is the averaging time, $\tilde{v}_{c,j} \in \mathbb{R}^{h_c}$ are the normalized spatial fields comprising s_c, $\tilde{V}_{c,j}$ the components of the state S_c prior to normalization, n_c is the number of fields observed in configuration x_c, and h_c is the number of degrees of freedom of each field. As an example, the first configuration's observed state S_1 may include as fields atmospheric soundings of temperature and specific humidity ($n_1 = 2$) measured at h_1 vertical locations above the surface, and the second configuration's state S_2 may include these fields as well as horizontal velocity profiles, measured at h_2 different locations. Normalization of the observed state S_c is performed using the pooled time variance $\sigma_{c,j}^2$ of each field $\tilde{V}_{c,j}$,

$$\tilde{v}_{c,j} = \sigma_{c,j}^{-1}\tilde{V}_{c,j}, \qquad \sigma_{c,j}^2 = h_c^{-1}\mathrm{tr}\Big[\mathrm{Cov}(\tilde{V}_{c,j})\Big], \tag{9}$$

where covariances are computed over a time interval $t_c \geq T_c$ following the heuristic of Section 2.1 to capture the expected magnitude of the data mismatch,

$$\mathrm{Cov}(\tilde{V}_{c,j}) = \frac{1}{t_c}\int_0^{t_c} \tilde{V}_{c,j}\tilde{V}_{c,j}^T d\tau - \frac{1}{t_c^2}\Big(\int_0^{t_c} \tilde{V}_{c,j}d\tau\Big)\Big(\int_0^{t_c} \tilde{V}_{c,j}d\tau\Big)^T. \tag{10}$$

We resort to pooled normalization, instead of normalizing each of the dimensions of the observed state S_c by their standard deviation, because some of the dimensions of the spatial fields $\tilde{V}_{c,j}$ may be unaffected by a given forcing. For example, in the atmospheric boundary layer, observations of liquid water specific humidity will always be zero below the lifting condensation level.

Stacking the observations from all configurations together, the full observation vector $\tilde{y}$ appearing in the global inverse problem (3) is

$$\tilde{y} = \begin{bmatrix} \tilde{y}_1 \\ \dots \\ \tilde{y}_{|C|} \end{bmatrix} \in \mathbb{R}^{\tilde{d}}, \qquad \tilde{d} = \sum_{c=1}^{|C|}\tilde{d}_c = \sum_{c=1}^{|C|} n_c h_c. \tag{11}$$

Following again the heuristic in Section 2.1, the noise covariance associated with each observation vector $\tilde{y}_c$ is $\tilde{\Gamma}_c = \mathrm{Cov}(s_c)$, computed as in equation (10). Given that the noise is constructed over configurations, the observational noise covariance is the block diagonal matrix

$$\tilde{\Gamma} = \begin{bmatrix} \tilde{\Gamma}_1 & & 0 \\ & \ddots & \\ 0 & & \tilde{\Gamma}_{|C|} \end{bmatrix} \in \mathbb{R}^{\tilde{d},\tilde{d}}, \qquad \tilde{\Gamma}_c = \mathrm{Cov}(s_c) \in \mathbb{R}^{\tilde{d}_c,\tilde{d}_c}, \tag{12}$$

where $\tilde{\Gamma}_c$ is the observational covariance matrix of configuration c, corresponding to data $\tilde{y}_c \in \mathbb{R}^{\tilde{d}_c}$.

2.2.3 Observations in a reduced space

Each covariance matrix $\tilde{\Gamma}_c$, possibly associated with high-dimensional observations and a finite sampling interval, is likely to be approximate rank-deficient and have a large condition number $\kappa = \sigma_1^2/\sigma_r^2$, where σ_i^2 is the i-th largest eigenvalue and r is the approximate rank of the matrix (Hansen, 1998). Rank-deficient problems arise when $\tilde{d}_c$ is

greater than or equal to the number of samples used to construct $\tilde{\Gamma}_c$, or when there exist eigenvalues σ_i^2 such that $\sigma_i^2/\sigma_1^2 \lesssim \epsilon_m$, where ϵ_m is a measure of data or machine precision.

An efficient regularization method for rank-deficient problems is to project the data from each configuration onto a lower dimensional encoding. If the lower dimensional encoding is obtained through principal component analysis (PCA),

$$y_c = P_c^T \tilde{y}_c, \qquad \Gamma_c^\dagger = P_c^T \tilde{\Gamma}_c P_c, \tag{13}$$

where $y_c \in \mathbb{R}^{d_c}$, P_c is the projection matrix formed by the d_c leading eigenvectors of $\tilde{\Gamma}_c$, and d_c should be chosen such that $d_c \leq r_c \leq \tilde{d}_c$, where r_c is the approximate rank of $\tilde{\Gamma}_c$. The actual value of d_c may be chosen through the discrepancy principle, generalized cross validation, or based on the preservation of a given fraction of the total variance, among other criteria (Reichel & Rodriguez, 2013). Projection (13) enables the use of the domain-agnostic $\tilde{y}$ by regularizing the associated inverse problem and lowering its computational cost. It also allows extending the observation vector to include linearly dependent data after appropriate normalization, such that $\tilde{y}_c$ in expression (8) may include normalized time integrals of all observed fields. Furthermore, since $\tilde{\Gamma}$ in (12) is block diagonal, the eigenvalue problem can be solved in parallel for different configurations. Note that projection (13) maximizes the projected variance for each configuration; it is different than performing PCA on $\tilde{\Gamma}$ in that it does not discriminate based on the total variance of each configuration. Disparities between the two approaches are further discussed in Appendix A.

Although projection (13) regularizes each $\tilde{\Gamma}_c$, the resulting global covariance matrix may be ill-conditioned if truncation is performed between eigenvalues that are close in value, or if the range of configuration variances $\mathrm{tr}(\tilde{\Gamma}_c)$ is large (Hansen, 1990). In this case, Tikhonov regularization can be used to limit the condition number κ of the global covariance matrix. The regularized projection can then be written as

$$y_c = P_c^T \tilde{y}_c, \qquad \Gamma_c = d_c P_c^T \tilde{\Gamma}_c P_c + \kappa_*^{-1} \sigma_1^2 I_{d_c}, \tag{14}$$

where κ_* is the limiting condition number of the global covariance matrix, σ_1^2 is the leading eigenvalue of the unregularized global covariance and I_{d_c} is the identity matrix. The condition number should be chosen to be $\kappa_* < \epsilon_m^{-1/2}$. In (14), since the number of retained principal modes may be different among configurations for a given truncation criterion, each block covariance matrix is scaled by d_c. Finally, the observation vector and noise covariance matrix read

$$y = \begin{bmatrix} y_1 \\ \dots \\ y_{|C|} \end{bmatrix} \in \mathbb{R}^d, \qquad \Gamma = \begin{bmatrix} \Gamma_1 & & 0 \\ & \ddots & \\ 0 & & \Gamma_{|C|} \end{bmatrix} \in \mathbb{R}^d, \tag{15}$$

which define a regularized inverse problem of the form (3). A schematic of the inverse problem construction process is given in Figure 1. The construction of y_c from each dynamical system configuration $\zeta(x_c)$ defines the observational map $\mathcal{H}_c$, used to obtained the forward model evaluation $\mathcal{G}_c(\cdot)$ from the dynamical model. The construction of each (y_c, Γ_c) pair, and the evaluation of $\mathcal{G}_c(\cdot)$, can be done in parallel.

2.3 Loss function

Given the observations constructed through equations (11)–(15), the solution θ^* to the inverse problem (3) is the minimizer of the loss function

$$L(\theta; y) = \frac{1}{|C|} ||y - \mathcal{G}(\theta)||_\Gamma^2 = \frac{1}{|C|} \sum_{c=1}^{|C|} L(\theta; y_c) = \frac{1}{|C|} \sum_{c=1}^{|C|} ||y_c - \mathcal{G}_c(\theta)||_{\Gamma_c}^2, \tag{16}$$

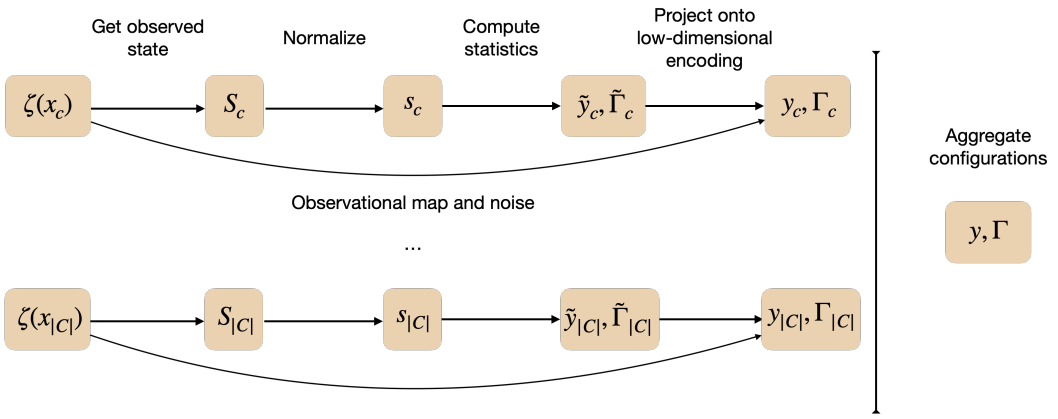

Figure 1: Schematic of the strategy used to construct a regularized inverse problem from observations of a dynamical system ζ. From left to right: (a) the observed state is obtained following Section 2.2.1 or from any observable fields; (b) the observed state is normalized; (c) mean and covariance of the normalized state are computed; (d) $\tilde{y}_c$ and $\tilde{\Gamma}_c$ are projected onto a lower dimension and regularized; (e) the statistical summaries of each configuration are aggregated, defining the global inverse problem (3).

that is,

$$\theta^* = \arg\min_{\theta} L(\theta; y), \tag{17}$$

where $||\cdot||_\Gamma$ denotes the Mahalanobis norm $\langle \cdot, \Gamma^{-1} \cdot \rangle$. The optimum θ^* arises as the maximum a posteriori (MAP) estimator in the Bayesian formulation of the inverse problem (3) using an uninformative prior (Kovachki & Stuart, 2019). The loss (16) represents the average configuration data misfit, and it is equivalent to using an error covariance $|C| \cdot \Gamma$. Although the $|C|^{-1}$ scaling may be regarded as a nuisance for minimization, it enables the use of mini-batch surrogates of the total loss in the calibration process.

The regularizing effect of projection (14) becomes apparent when the gradient of the loss (16) with respect to the forward map $\mathcal{G}$ is considered,

$$\nabla L(\theta; y) \propto (D\mathcal{G}(\theta))^T \Gamma^{-1} (\mathcal{G}(\theta) - y). \tag{18}$$

Here, $D\mathcal{G}(\theta) \in \mathbb{R}^{d \times p}$ is the Jacobian matrix of $\mathcal{G}$ evaluated at θ. Projection (14) regularizes the linear system $\Gamma^{-1}(\mathcal{G}(\theta) - y)$ in expression (18). This is crucial for convergence with gradient-based optimization methods. Although the ensemble Kalman algorithms presented in Section 3 do not compute the gradient (18) explicitly, they do rely on approximations of it, so this regularization effect still applies.

2.3.1 Mini-batch loss evaluations

Iterative optimization methods require the evaluation of $L(\theta; y)$ at each iteration, which entails evaluating the dynamical model $\Psi(\phi)$ in all configurations C and can be very computationally demanding. A less onerous alternative is to evaluate the loss for a mini-batch of configurations $B \subset C$ at each iteration,

$$L(\theta; y_B) = \frac{1}{|B|} \sum_{c=1}^{|B|} ||y_c - \mathcal{G}_c(\theta)||^2_{\Gamma_c}, \tag{19}$$

and use $L(\theta; y_B)$ to update θ instead. The use of $L(\theta; y_B)$ in lieu of $L(\theta; y)$ may be regarded as using noisy evaluations of the total loss for each parameter update. As in equation (16), the mini-batch loss (19) represents the average configuration data misfit. To

estimate the total data misfit over C, we would multiply expressions (16) and (19) by $|C|$.

Mini-batch optimization is widely employed in the field of deep learning, where it has been shown to help avoid convergence to sharp minima that generalize poorly (M. Li et al., 2014; Keskar et al., 2016). Understanding the behavior of optimizers when using mini-batches is crucial for online learning, where observations become available sequentially and the total loss (16) cannot be sampled. Moreover, it provides insight into the appropriateness of training sequentially on seasonal or geographically sparse data in Earth system modeling applications. We explore the effect of mini-batching on the solution of the inverse problem in Section 4.2, training sequentially on randomly sampled configurations with markedly different dynamics.

3 Ensemble Kalman methods for optimization

We consider two highly-parallelizable gradient-free optimization algorithms based on the extended Kalman filter: ensemble Kalman inversion (EKI, Iglesias et al., 2013) and unscented Kalman inversion (UKI, Huang et al., 2022). Both algorithms draw heavily on Gaussian conditioning for their derivation, such that underlying their update rules is the approximation of the parameter distribution as Gaussian (Huang et al., 2022).

EKI seeks a solution to the inverse problem (3) by evolving an ensemble of J parameter vectors $\theta^{(j)} \in \mathbb{R}^p$, which is used to obtain empirical estimates of covariances in parameter space at each step of the algorithm. UKI instead relies on deterministic quadrature rules for covariance estimation, using $2p+1$ parameter vectors in each iteration. Both methods have been used succesfully in a wide variety of inverse problems (Cleary et al., 2021; Huang et al., 2022). We demonstrate them here in the context of training models that may experience numerical instabilities for a priori unknown parameter combinations, starting with a brief review of the algorithms.

3.1 Ensemble Kalman inversion (EKI)

Ensemble Kalman inversion searches for an optimal solution (17) to the inverse problem (3) through iterative updates of an initial parameter ensemble $\Theta_0 = [\theta_0^{(1)}, \dots, \theta_0^{(J)}]$. This initial ensemble is taken to be randomly sampled from a Gaussian prior $\mathcal{N}(m_0, \Sigma_0)$ in parameter space. The EKI update equation for the ensemble at iteration n is (Schillings & Stuart, 2017)

$$\Theta_{n+1} = \Theta_n + \mathrm{Cov}(\theta_n, \mathcal{G}_n)\left[\mathrm{Cov}(\mathcal{G}_n, \mathcal{G}_n) + \Delta t^{-1}\Gamma\right]^{-1} \varepsilon(\Theta_n), \tag{20}$$

where $\Theta_n \in \mathbb{R}^{p\times J}$, Δt is a nominal learning rate of the algorithm, and $\varepsilon(\Theta_n) \in \mathbb{R}^{d\times J}$ encodes the mismatch between the forward model evaluations and the data,

$$\varepsilon(\Theta_n) = [y_{n+1}^{(1)} - \mathcal{G}(\theta_n^{(1)}), \dots, y_{n+1}^{(J)} - \mathcal{G}(\theta_n^{(J)})], \tag{21}$$

where

$$y_{n+1}^{(j)} = y + \xi_{n+1}^{(j)}, \qquad \xi_{n+1}^{(j)} \sim \mathcal{N}(0, \Delta t^{-1}\Gamma). \tag{22}$$

All covariances in (20) are estimated as sample covariances from the J ensemble members,

$$\mathrm{Cov}(\theta_n, \mathcal{G}_n) = \frac{1}{J}\Big(\Theta_n - \frac{1}{J}\sum_j \theta_n^{(j)}\mathbf{1}^T\Big)\Big(\mathcal{G}_{\Theta_n} - \frac{1}{J}\sum_j \mathcal{G}(\theta_n^{(j)})\mathbf{1}^T\Big)^T, \tag{23}$$

$$\mathrm{Cov}(\mathcal{G}_n, \mathcal{G}_n) = \frac{1}{J}\Big(\mathcal{G}_{\Theta_n} - \frac{1}{J}\sum_j \mathcal{G}(\theta_n^{(j)})\mathbf{1}^T\Big)\Big(\mathcal{G}_{\Theta_n} - \frac{1}{J}\sum_j \mathcal{G}(\theta_n^{(j)})\mathbf{1}^T\Big)^T, \tag{24}$$

where $\mathcal{G}_{\Theta_n} = [\mathcal{G}(\theta_n^{(1)}), \dots, \mathcal{G}(\theta_n^{(J)})]$, and $\mathbf{1} \in \mathbb{R}^J$ is the all-ones vector. Note that the sample covariances (23) and (24) have at most ranks $\min(\min(d,p), J-1)$ and $\min(d, J-$

1), respectively. From definitions (14) and (15), $\text{rank}(\Gamma) = d$ by construction, so the linear system in (20) is well-defined even for $J < d$.

Through iterative application of the update equation (20), the ensemble Θ minimizes the projection of the model-data mismatch on the linear span of its members. This emphasizes the importance of using $J > p$ ensemble members to span the whole parameter space. In this study, we limit the use of EKI and UKI to the calibration of dynamical models for which $J \sim p$ is feasible. For models with a large number of parameters, localization techniques can be used to maintain performance with $J \ll p$ (Tong & Morzfeld, 2022).

The update rule (20) drives the ensemble toward consensus, in the sense that $|\text{Cov}(\theta_n, \mathcal{G}_n)| \to 0$ as $n \to \infty$. This collapse property precludes obtaining information about parameter uncertainties directly from EKI. However, the sequence of parameter-output pairs $\{\Theta_n, \mathcal{G}_{\Theta_n}\}$ can be used to train emulators for uncertainty quantification (Cleary et al., 2021).

3.1.1 Addressing model failures within the ensemble

For some models $\Psi(\cdot)$, we may not know a priori the parameter space region U for which trajectories remain physical or numerically stable. For instance, the Courant–Friedrichs–Lewy condition in parameterized fluid solvers may change nonlinearly with model parameters, or the initialized weights from a non-interpretable neural network parameterization may lead to unstable trajectories. In such situations, we need to modify update (20) to account for model failures within the ensemble.

Here we propose a failsafe EKI update based on the successful parameter ensemble. Let $\Theta_{s,n} = [\theta_{s,n}^{(1)}, \ldots, \theta_{s,n}^{(J_s)}]$ be the successful ensemble, for which each model $\Psi(\theta_{s,n}^{(j)})$ provides physical trajectories, and let $\theta_{f,n}^{(k)}$ be the ensemble members for which the model $\Psi(\theta_{f,n}^{(k)})$ fails. We update the successful ensemble $\Theta_{s,n}$ to $\Theta_{s,n+1}$ using expression (20), and each failed ensemble member as

$$\theta_{f,n+1}^{(k)} \sim \mathcal{N}\left(m_{s,n+1}, \Sigma_{s,n+1}\right), \tag{25}$$

where

$$m_{s,n+1} = \frac{1}{J_s}\sum_{j=1}^{J_s}\theta_{s,n+1}^{(j)}, \qquad \Sigma_{s,n+1} = \text{Cov}(\theta_{s,n+1}, \theta_{s,n+1}) + \kappa_*^{-1}\sigma_{s,1}^2 I. \tag{26}$$

Here, κ_* is a limiting condition number and $\sigma_{s,1}^2$ is the largest eigenvalue of the sample covariance $\text{Cov}(\theta_{s,n+1}, \theta_{s,n+1})$. This update has proved very effective for us in practice, even in situations where $J_s < J/2$, and is used throughout Section 4. It may be combined with other conditioning techniques at initialization. For instance, the initial ensemble Θ_0 may be drawn recursively from the prior $\mathcal{N}(m_0, \Sigma_0)$ until the number of failed members is reduced below an acceptable threshold.

3.2 Unscented Kalman inversion (UKI)

The UKI algorithm updates estimates of the mean and covariance of the parameter distribution, initialized from an initial guess $\mathcal{N}(m_0, \Sigma_0)$. Several variants of the algorithm have been developed, with different properties (Huang et al., 2022). In this article, we employ the update rules

$$m_{n+1} = m_n + \text{Cov}_q(\theta_n, \mathcal{G}_n)\left[\text{Cov}_q(\mathcal{G}_n, \mathcal{G}_n) + 2\Delta t^{-1}\Gamma\right]^{-1}\varepsilon(m_n), \tag{27}$$

$$\Sigma_{n+1} = (1+\Delta t)\Sigma_n - \text{Cov}_q(\theta_n, \mathcal{G}_n)\left[\text{Cov}_q(\mathcal{G}_n, \mathcal{G}_n) + 2\Delta t^{-1}\Gamma\right]^{-1}\text{Cov}_q(\theta_n, \mathcal{G}_n)^T, \tag{28}$$

where m_n and Σ_n are the estimates of the parameter mean and covariance after n iterations of the algorithm, and $\varepsilon(m_n) = y - \mathcal{G}(m_n)$ is the data-model mismatch of the

mean prediction. The covariances $\mathrm{Cov}_q(\theta_n, \mathcal{G}_n)$ and $\mathrm{Cov}_q(\mathcal{G}_n, \mathcal{G}_n)$ in (27) and (28) are computed through quadratures over $2p+1$ sigma points defined as

$$\hat{\theta}_n^{(j)} = m_n + a\sqrt{p}[\sqrt{\Sigma_n(1+\Delta t)}]_j, \qquad 1 \le j \le p, \tag{29}$$
$$\hat{\theta}_n^{(j+p)} = m_n - a\sqrt{p}[\sqrt{\Sigma_n(1+\Delta t)}]_j, \qquad 1 \le j \le p,$$

where $[\sqrt{\Gamma}]_j$ is the j-th column of the Cholesky factor of Γ, $a = \min(\sqrt{4/p}, 1)$ is a hyperparameter defined in Huang et al. (2022), and $\hat{\theta}_n^{(0)} = m_n$ is the central sigma point. The quadratures are then defined as

$$\mathrm{Cov}_q(\theta_n, \mathcal{G}_n) = \sum_{j=1}^{2p} w_j(\hat{\theta}_n^{(j)} - m_n)(\mathcal{G}(\hat{\theta}_n^{(j)}) - \mathcal{G}(m_n))^T, \tag{30}$$

$$\mathrm{Cov}_q(\mathcal{G}_n, \mathcal{G}_n) = \sum_{j=1}^{2p} w_j(\mathcal{G}(\hat{\theta}_n^{(j)}) - \mathcal{G}(m_n))(\mathcal{G}(\hat{\theta}_n^{(j)}) - \mathcal{G}(m_n))^T, \tag{31}$$

where w_j are the quadrature weights,

$$w_j = (2a^2 p)^{-1}, \qquad j \ge 1. \tag{32}$$

In contrast to EKI, the update equations of UKI are deterministic given an initial guess $\mathcal{N}(m_0, \Sigma_0)$. A limitation of this algorithm is that the number of sigma points scales linearly with p, which precludes its use when training models with a large number of parameters. However, for situations where using an ensemble of $2p+1$ members is tractable, UKI improves upon EKI by providing information about parameter uncertainty. UKI does not drive the $2p+1$ parameter vectors toward consensus; their relative location is defined by the covariance Σ_n. The particular variant of UKI used here ensures that the steady-state estimate of Σ_n in the limit $n \to \infty$ converges towards an estimate of the parametric error covariance matrix, given $d \ge p$ (Huang et al., 2022),

$$\Sigma_\infty \approx \mathrm{Cov}_q(\theta_\infty, \mathcal{G}_\infty)\left[\Delta t \cdot \mathrm{Cov}_q(\mathcal{G}_\infty, \mathcal{G}_\infty) + 2\Gamma\right]^{-1} \mathrm{Cov}_q(\theta_\infty, \mathcal{G}_\infty)^T. \tag{33}$$

As shown next, the condition $d > p$ is satisfied by construction when L_2 regularization is added to UKI. The fact that the limit (33) does not depend on Σ_0 has two important consequences. On one hand, it precludes the interpretation of $\mathcal{N}(m_0, \Sigma_0)$ as a Bayesian prior. On the other hand, this avoids the need to find a *wide enough* prior in parameter space, which can prove difficult for parameters θ without physical interpretation, and tends to increase the fraction of model failures within the ensemble. For parameters for which a Bayesian interpretation is considered beneficial, a prior can still be enforced through L_2 regularization. A modification of the UKI dynamics robust to model failures, similar to the one proposed for EKI, is discussed in Appendix B.

3.3 L_2 regularization in ensemble Kalman methods

The EKI algorithm implicitly regularizes the inverse problem by searching for the optimal solution (17) over the finite-dimensional space spanned by the initial ensemble. Although the UKI algorithm does not share this property, both the EKI and UKI algorithms described in Sections 3.1 and 3.2 can be equipped with L_2 regularization by considering the augmented inverse problem (Chada et al., 2020)

$$\begin{bmatrix} y \\ m_p \end{bmatrix} = \begin{bmatrix} \mathcal{G}(\theta) \\ \theta \end{bmatrix} + \begin{bmatrix} \delta + \eta \\ \lambda \end{bmatrix}, \tag{34}$$

where $m_p \in \mathbb{R}^p$ is the parameter prior mean, $\lambda \sim \mathcal{N}(0, \Lambda)$ is artificial noise in the parameters θ, and Λ is the covariance matrix that defines the degree of regularization in parameter space. The solution to the inverse problem (34) then satisfies

$$\theta^* = \arg\min_\theta \Big[L(\theta; y) + ||\theta - m_p||_\Lambda^2\Big]. \tag{35}$$

A Bayesian perspective to the optimization problem suggests the use of the prior variance to define the regularizer Λ. This perspective is particularly interesting for the UKI algorithm, which provides estimates of the parameter sensitivities in the calibration process (Huang et al., 2022).

4 Application to an atmospheric subgrid-scale model

In this section, the framework and algorithms discussed in Sections 2 and 3 are used to learn closure parameters within an EDMF scheme of atmospheric turbulence and convection. The EDMF scheme is derived by spatially filtering the Navier-Stokes equations for an anelastic fluid, and then decomposing the subgrid flow into $n > 1$ distinct subdomains with potentially moving boundaries (Tan et al., 2018; Cohen et al., 2020). We retain second-order moments for one of the subdomains, the environment. Covariances within the other subdomains (updrafts) are neglected, which circumvents the need for turbulence closures therein. In the end, the EDMF equations require closure for the turbulent diffusivity and dissipation in the environment, and the mass, momentum, and tracer fluxes between environment and updrafts. In what follows, we consider an EDMF scheme with a single updraft.

We consider the EDMF scheme discussed in Cohen et al. (2020); Lopez-Gomez et al. (2020); He et al. (2021), which is implemented in a single-column model (SCM). Within this SCM, we first seek to learn 16 closure parameters: 5 describing turbulent mixing, dissipation, and mixing inhibition by stratification (Lopez-Gomez et al., 2020), 3 describing the momentum exchange between subdomains (He et al., 2021), 7 describing entrainment and detrainment fluxes from the updrafts and into the environment (Cohen et al., 2020), and another one defining the surface area fraction occupied by updrafts. In Section 4.4, we substitute the empirical dynamical entrainment/detrainment closure proposed in Cohen et al. (2020) by a neural network and train the resulting physics-based machine-learning model.

The name, prior range U, and reference to the definition of each parameter in the literature are given in Table 1. The prior mean is taken to be equal to the parameter values used in Lopez-Gomez et al. (2020); Cohen et al. (2020). The prior in unconstrained space necessary to initialize the calibration algorithms, $\mathcal{N}(m_0, \Sigma_0)$, is obtained from the prior mean and range through the use of a transformation $\mathcal{T} : U \to \mathbb{R}^p$ defined in Appendix C. In all cases, we employ the failsafe modifications of the EKI and UKI algorithms (Section 3.1.1 and Appendix B) equipped with regularization, solving the augmented inverse problem (35) with $m_p = m_0$ and $\Lambda = I$, unless otherwise specified. This regularization is consistent with the prior in Table 1 and the transformation $\mathcal{T}$ used for the parameters.

4.1 Description of LES data and model configurations

The data used for training and testing the EDMF scheme are taken from the LES library described in Shen et al. (2022). This library contains high-resolution simulations of low-level clouds spanning the stratocumulus-to-cumulus transition in the East Pacific Ocean. The large-scale forcing used for these simulations is derived from the cfSites output of the HadGEM2-A model, retrieved from the Coupled Model Intercomparison Project Phase 5 (CMIP5) archive. In particular, the monthly climatology of the cfSites output is computed over the 5-year period 2004-2008, and used to initialize and force large-eddy simulations for a period of 6 days. Radiative forcing is computed interactively using the Rapid Radiative Transfer Model (RRTM, Mlawer et al., 1997).

The SCM runs are initialized from the coarse-grained LES fields after 5.75 days of simulation, and run for 6 hours. This runtime was chosen to be much longer than the equilibration time of the SCM to the steady forcing; experiments using a runtime of 12

Table 1: Parameters ϕ considered for calibration in this study. The prior mean values are taken from LG2020 (Lopez-Gomez et al., 2020), C2020 (Cohen et al., 2020) and H2021 (He et al., 2021), where a physical description of the parameters may be found.

Symbol	Description	Prior range	Prior mean
c_m	Eddy viscosity coefficient	$(0.01, 1.0)$	0.14, LG2020
c_d	Turbulent dissipation coefficient	$(0.01, 1.0)$	0.22, LG2020
c_b	Static stability coefficient	$(0.01, 1.0)$	0.63, LG2020
$\mathrm{Pr}_{t,0}$	Neutral turbulent Prandtl number	$(0.5, 1.5)$	0.74, LG2020
κ_*	Ratio of rms turbulent velocity to friction velocity	$(1.0, 4.0)$	1.94, LG2020
c_ε	Entrainment rate coefficient	$(0, 1)$	0.13, C2020
c_δ	Detrainment rate coefficient	$(0, 1)$	0.51, C2020
c_γ	Turbulent entrainment rate coefficient	$(0, 10)$	0.075, C2020
β	Detrainment relative humidity power law	$(0, 4)$	2.0, C2020
μ_0	Entrainment sigmoidal activation parameter	$(10^{-6}, 10^{-2})$	$4 \cdot 10^{-4}$, C2020
χi	Updraft-environment buoyancy mixing ratio	$(0, 1)$	0.25, C2020
c_λ	Turbulence-induced entrainment coefficient	$(0, 10)$	0.3, C2020
a_s	Updraft surface area fraction	$(0.01, 0.5)$	0.1, C2020
α_b	Updraft virtual mass loading coefficient	$(0, 10)$	0.12, H2021
α_a	Updraft advection damping coefficient	$(0, 100)$	0.001, H2021
α_d	Updraft drag coefficient	$(0, 50)$	10.0, H2021

hours only resulted in a doubling of the forward model computational cost. Large-scale forcing is identical to that of the LES, and the radiative heating rates are given by the horizontal mean of the rates experienced by the high-resolution simulations. The observational map used to define the inverse problem follows the guidelines of Section 2.2, using time and horizontally averaged vertical profiles from the last $T_c = 3$ hours of simulation, at a vertical resolution of $\Delta z = 50$ m. Following the notation in Section 2.2, we consider the state

$$S_c = [\bar{u}, \bar{v}, \bar{s}, \bar{q}_l, \bar{q}_t, \overline{w'q_t'}, \overline{w's'}]^T, \tag{36}$$

where $\bar{(\cdot)}$ denotes time and horizontal averaging, $\bar{u}$ and $\bar{v}$ are the horizontal velocity components, $\bar{s}$ is the entropy, $\bar{q}_t$ is the total specific humidity, $\overline{w'q_t'}$ and $\overline{w's'}$ are vertical fluxes of moisture and entropy, and $\bar{q}_l$ is the liquid water specific humidity. The pooled variances for normalization and the covariance matrix Γ are obtained from the full 6 day statistics of the large-eddy simulations to capture the internal variability of the system. Finally, a low-dimensional encoding is obtained from the state vector (36) through truncated PCA, truncating the dimension of the noise covariance matrix so as to preserve 99% of the total noise variance. Calibration results using fewer observed fields at a coarser resolution are discussed in Section 4.3.

In the training data we include a total of 60 LES configurations from the Atmospheric Model Intercomparison Project (AMIP) experiment, spanning the months of January, April, July and October. Results are also shown for a validation set, which includes January and July simulations from the AMIP4K experiment, where sea surface temperature is increased by 4 K with respect to AMIP (Shen et al., 2022). Validation results are representative of the generalizability of the trained model for the simulation of a warming climate; the model was not trained on these warmer conditions.

4.2 Calibration using mini-batch loss evaluations

To demonstrate the effectiveness of Kalman inversion in practical settings where evaluating all configurations of interest per iteration may be too expensive or impossible (e.g., due to sequential data availability), we present calibration results using mini-batches. Batching introduces noise in the loss evaluations due to sampling error. For this reason, the behavior of Kalman inversion algorithms using mini-batches is representative of their robustness to other sources of noise, such as noise in the data or stochasticity of the dynamical model. Sampling noise also has implications for uncertainty quantification with UKI, since additional noise leads to a larger uncertainty estimate Σ_n. If we are interested in capturing the uncertainty given the full training set, we can correct for the sampling error by using $\Delta t = |C|/|B|$, which effectively reduces Γ in updates (20) and (27). This is the approach we take in this work.

For training, data are fed to the algorithm by drawing $|B|$ configurations randomly and without replacement from the training set at every iteration. Configurations are reshuffled at the end of every epoch (i.e., every full pass through the training set). Figure 2 shows the evolution of the training and validation errors for UKI and EKI, using training batches of 5 and 20 configurations; the dependence of EKI results on ensemble size is explored in Section 4.4. Since the total number of configurations in the training set is 60, an epoch requires 12 iterations when using $|B| = 5$ and 3 when using $|B| = 20$. For many geophysical applications, the cost of evaluating an ensemble of long-term statistics $\mathcal{G}(\cdot)$ from a forward model is significantly higher than performing the inversion updates (20) or (27). In these situations, a training epoch has similar computational overhead for any value of $|B|$.

The training error is evaluated here in normalized physical space with respect to the current batch,

$$\mathrm{MSE}(\theta; \tilde{y}_B) = \frac{1}{\tilde{d}_B} ||\tilde{y}_B - \tilde{\mathcal{G}}_B(\theta)||^2, \tag{37}$$

where $\tilde{y}_B \in \mathbb{R}^{\tilde{d}_B}$. The validation error is defined similarly, but it is computed over the entire validation set at every iteration. Thus, variations in the validation error are only due to changes in the model parameters; there is no random data sampling. The training and validation errors decrease sharply during the first epoch. Subsequent epochs fine-tune the model parameters, further reducing the data-model mismatch. It is remarkable and important that the validation error decreases by about the same magnitude as the training error, demonstrating that the parameterization approach that leverages a physical model generalizes successfully out of the present-climate training sample to a warmer climate.

Both EKI and UKI display efficient training in the low batch-size regime: the validation error tends to be lower for smaller batches after a fixed number of epochs. Hence, decreasing batch size in EKI and UKI can help reduce the computational cost of model calibration. The optimal batch size will depend on the CPU and wall-clock time constraints of the user. Although using smaller batches reduces CPU time, it requires more serial operations, so using larger batches can reduce wall-clock time.

The sampling noise due to the use of different configurations (e.g., stratocumulus versus cumulus regimes) increases for smaller batches. Although both algorithms achieve convergence for a wide range of batch sizes, we find that EKI is more robust than UKI to high levels of noise. This is shown in the inset of Figure 2b for $|B| = 5$, and in Appendix D for $|B| = 1$. Other differences between UKI and EKI may be observed in Figure 2. The consensus property of EKI leads to a collapse of the model error spread after a few iterations, converging to a point estimate. On the other hand, the UKI ensemble converges to an MSE spread characteristic of the parameter space region defined by the distribution $\mathcal{N}(m_n, \Sigma_n)$.

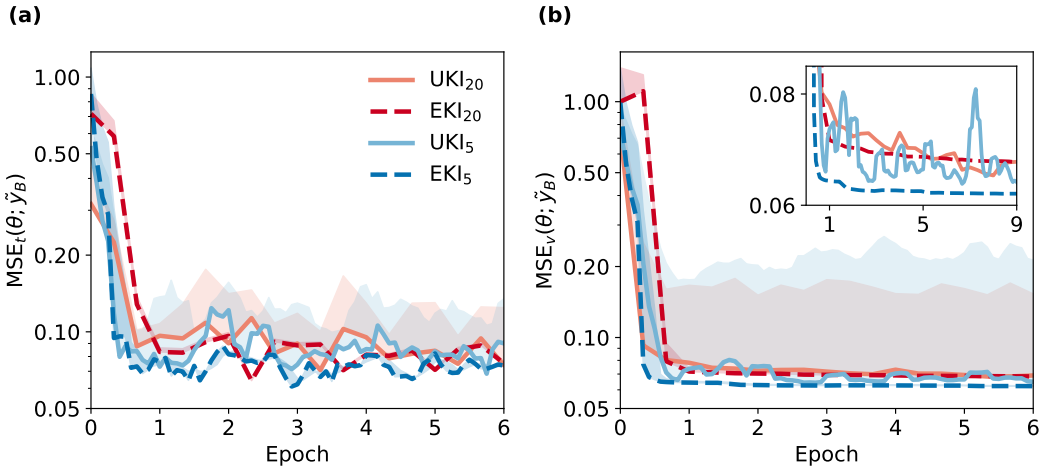

Figure 2: Batch (a) training and (b) validation mean squared error as defined in equation (37). Lines represent the error of the ensemble mean $\bar{\theta}$, MSE($\bar{\theta}$, $\tilde{y}_B$), and the shading represents the ensemble standard deviation of MSE(θ; $\tilde{y}_B$). All errors are normalized with respect to the largest initial MSE$_v$($\bar{\theta}$, $\tilde{y}_B$), so they can be compared. Results are shown for calibration with EKI and UKI, using $J = 2p + 1$ and training batch sizes $|B| = 5,\ 20$. Errors for $|B| = 5$ are averaged using a rolling mean of 20 configurations to enable comparison with $|B| = 20$. In (b), the inset focuses on the validation error evolution for a longer training period.

The evolution of the parameter estimate (m_n, Σ_n) is depicted in Figure 3 through the turbulent dissipation c_d, updraft advection damping α_a and surface area fraction a_s. The UKI estimate provides information about parameter uncertainty, whereas EKI only provides a point estimate (i.e., m_n). From the UKI estimate we can observe that the training set constrains the likely values of the turbulent dissipation (c_d) and surface area fraction (a_s) to a significantly smaller region than the prior. However, the magnitude of updraft advection damping (α_a) is not identifiable using this dataset. For non-identifiable parameters, the corresponding diagonal elements of Σ_n converge to the prior variance used in the regularized problem (34), as shown for α_a in Figure 3b.

The covariance estimate Σ_n also provides information about correlations between model parameters and total reduction of uncertainty, as shown in Figure 4. For the current stratocumulus-to-cumulus transition dataset, our EDMF model shows moderate correlations between parameters regulating the turbulence kinetic energy budget in the boundary layer (c_b, c_m, c_d, see Lopez-Gomez et al., 2020). We also find entrainment to be negatively correlated to surface updraft area fraction, detrainment and drag. These correlations can be used to improve parameterizations at the process level.

Vertical profiles of $\bar{q}_l, \overline{w'q'_t}$ and $\bar{u}$ from the validation set are compared to the reference LES profiles in Figure 5. The calibrated model yields smoother and more accurate profiles than the model before training. In particular, calibration significantly reduces biases in liquid water specific humidity and moisture transport for both stratocumulus and cumulus cloud regimes in the 4 K-warmer AMIP4K experiment. These results confirm that the dynamical model can be trained using a low-dimensional encoding of the time statistics, as proposed in Section 2. They also highlight the generalizability of sparse physics-based models.

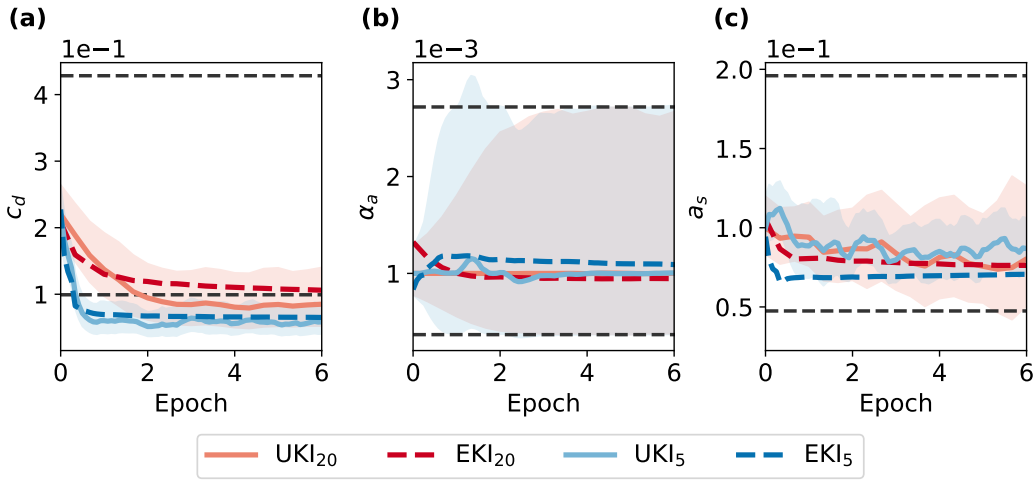

Figure 3: Evolution of the parameter estimate components corresponding to turbulent dissipation (c_d), updraft advection damping (α_a) and surface area fraction (a_s). All values are given in physical (constrained) space. The solid lines describe the trajectories of the mean estimate, $\mathcal{T}^{-1}(m_n)$. For UKI, the marginal $\pm\sigma$ uncertainty band is included in shading. This uncertainty is equal to $\pm\mathcal{T}^{-1}(\sqrt{(\Sigma_n)_{i,i}})$ for the i-th parameter. The black dashed lines are the $\pm\sigma$ uncertainty bands of the prior used for regularization. Legend as in Figure 2.

4.3 Calibration using partial observations

Another application of synthetic high-resolution data is the study of calibration sensitivity to data resolution and partial loss of information. Such sensitivity studies can inform the technical requirements of future observing systems or field campaigns (Suselj et al., 2020), and are easily implemented with ensemble and unscented Kalman inversion through modifications of the observational map $\mathcal{H}$.

Here, we employ the EKI and UKI algorithms for this task by using coarser training data at a vertical resolution of $\Delta z = 200$ m. In addition, we consider only a subset of fields for which real observational data may be obtained in practice: the liquid water potential temperature $\bar{\theta}_l$, the total water specific humidity $\bar{q}_t$ and the liquid water specific humidity $\bar{q}_l$ (National Academies of Sciences, Engineering, and Medicine, 2018; Suselj et al., 2020). Figure 6 compares calibration results using this reduced setup with the results obtained using the full high-resolution observations of Section 4.2. The loss of information is evident in the inability of the algorithms to find the same minimum reached with richer observations. Nevertheless, Kalman inversion reduces significantly the validation error from the prior even with sparser data and a limited number of fields.

The identifiability of individual parameters as a function of the observational map $\mathcal{H}$ can be inferred from the UKI Σ_n diagnostic. Figure 6 shows that the partial observations of temperature and humidity are enough to constrain the entrainment coefficient in the EDMF scheme considered. However, the loss of information with respect to the original observations leads to much poorer constraints on the turbulent dissipation coefficient. The same comparison can be performed for any parameter of interest to inform observational requirements to constrain models at the process level. This diagnostic is an important advantage of UKI over EKI; identifiability is not directly inferable from ensemble Kalman inversion due to the ensemble collapse. Nevertheless, this information can be recovered through the emulation of the forward map (Cleary et al., 2021).

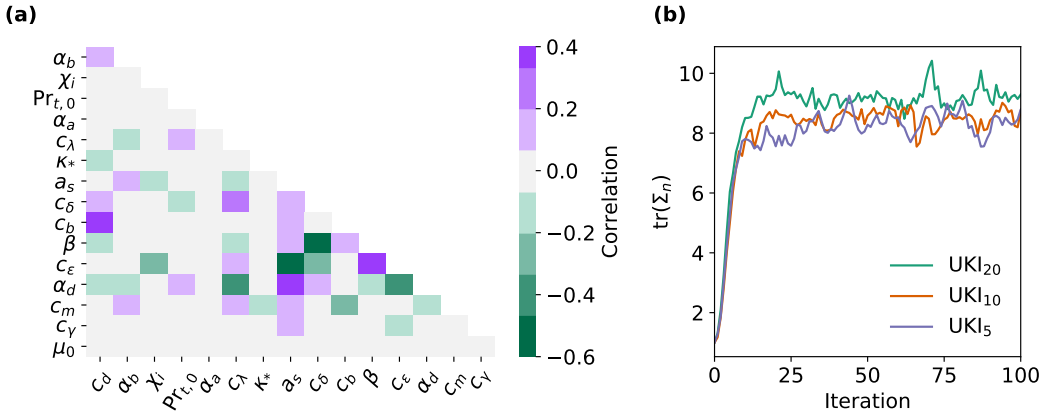

Figure 4: Parameter correlations estimated from UKI using $|B| = 20$ (a), and evolution of the total parameter variance from UKI using $|B| = 20$, 10 and 5. For comparison, the prior variance, encoded in UKI through the augmented system (34), is $\mathrm{tr}(\Lambda) = 16$. Note that the initial covariance estimate used in UKI (with $\mathrm{tr}(\Sigma_0) = 1$) is decoupled from the prior. Symbols follow Table 1.

The use of partial observations also highlights the benefits of learning from time statistics instead of tendencies. Learning from statistics not only ensures that the calibrated dynamical model is stable, which requires a leap of faith when training on instantaneous tendencies (Bretherton et al., 2022). It also couples the evolution of thermodynamic and dynamical fields, which can improve the forecast of fields unseen during training. An example is shown in Figure 7. The model calibrated using thermodynamic profiles improves upon the prior model in the forecast of horizontal velocities within the boundary and cloud layers. A common reason to use tendencies for calibration is the use of supervised learning techniques that are easy to implement for neural network architectures (e.g., Bretherton et al., 2022). In the next subsection, we demonstrate the power of UKI and EKI to calibrate hybrid models with embedded neural network parameterizations.

4.4 Calibration of a hybrid model with embedded neural network closures

We consider now a hybrid EDMF scheme that substitutes the dynamical entrainment and detrainment closures proposed by Cohen et al. (2020) with a three-layer dense neural network; see Cohen et al. (2020) for a review of how these terms affect the EDMF scheme dynamics. We define the fractional entrainment (ϵ) and detrainment (δ) rates as

$$\begin{bmatrix}\epsilon\\ \delta\end{bmatrix} = \frac{1}{z}\mathrm{NN}_3(\Pi_1, \dots, \Pi_6), \tag{38}$$

where z is the height, and the hidden layers of NN_3 have 5 and 4 nodes, from input to outputs. Our closure (38) seeks to learn local expressions for the z-normalized entrainment/detrainment rates, which have been shown to vary weakly in empirical studies of shallow cumulus convection (Siebesma, 1996; de Roode et al., 2000). The neural network inputs $\Pi_1, \dots, \Pi_6$ are 6 nondimensional groups on which entrainment and detrainment may depend, defined as

$$\Pi_1 = \frac{z(b_{up} - b_{en})}{(w_{up} - w_{en})^2 + w_d^2}, \tag{39a}$$

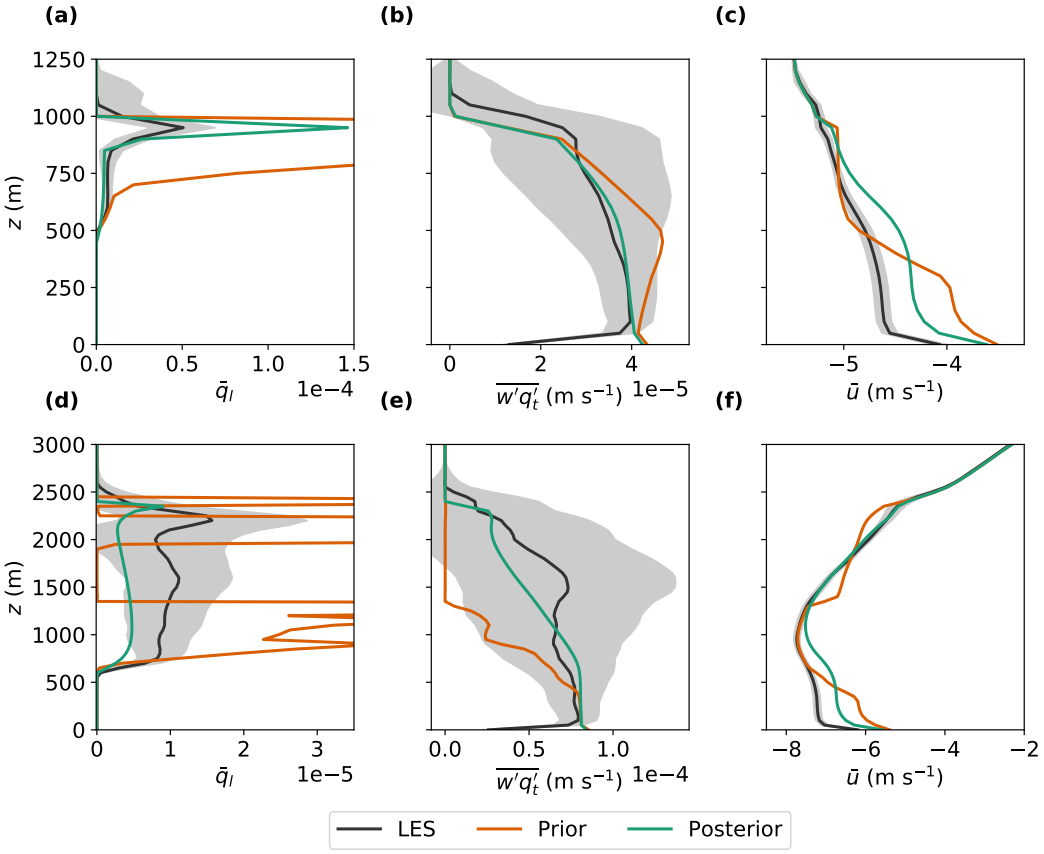

Figure 5: Prior, posterior and LES profiles of liquid water specific humidity ($\bar{q}_l$), subgrid-scale moisture flux ($\overline{w'q'_t}$) and zonal velocity ($\bar{u}$) for cfSites 5 (top) and 14 (bottom) using July forcing from the AMIP4K experiment as in Shen et al. (2022). The shading represents the internal variability of the LES simulations over 6 days of steady forcing, and the full lines represent 3-hour time-averaged profiles. Prior and posterior results are point estimates evaluated at the parameter vector closest to the ensemble mean of an EKI calibration process with $|B| = 5$ and $J = 2p + 1$.

$$\Pi_2 = \frac{a_{up}w_{up}^2 + (1-a_{up})w_{en}^2}{2(1-a_{up})e_{en} + a_{up}w_{up}^2 + (1-a_{up})w_{en}^2}, \tag{39b}$$

$$\Pi_3 = \sqrt{a_{up}}, \tag{39c}$$

$$\Pi_4 = \mathrm{RH}_{up} - \mathrm{RH}_{en}, \tag{39d}$$

$$\Pi_5 = z/H_{up}, \tag{39e}$$

$$\Pi_6 = gz/R_dT_{\mathrm{ref}}. \tag{39f}$$

In expressions (39), w_d is the Deardorff convective velocity, g is the gravitational acceleration, R_d is the ideal gas constant for dry air and T_{ref} is a reference temperature. The subscripts up and en denote updraft and environment, respectively. a_{up} is the updraft area fraction, H_{up} the updraft top height and e_{en} the environmental turbulence kinetic energy. The relative humidity RH, vertical velocity with respect to the grid mean w, and buoyancy b are defined for both updraft and environment.

The neural network closure (38) introduces 63 additional coefficients with respect to the entrainment and detrainment closure calibrated in Sections 4.2 and 4.3, for a total of 79 parameters. As the closure complexity increases, it is most practical to use EKI

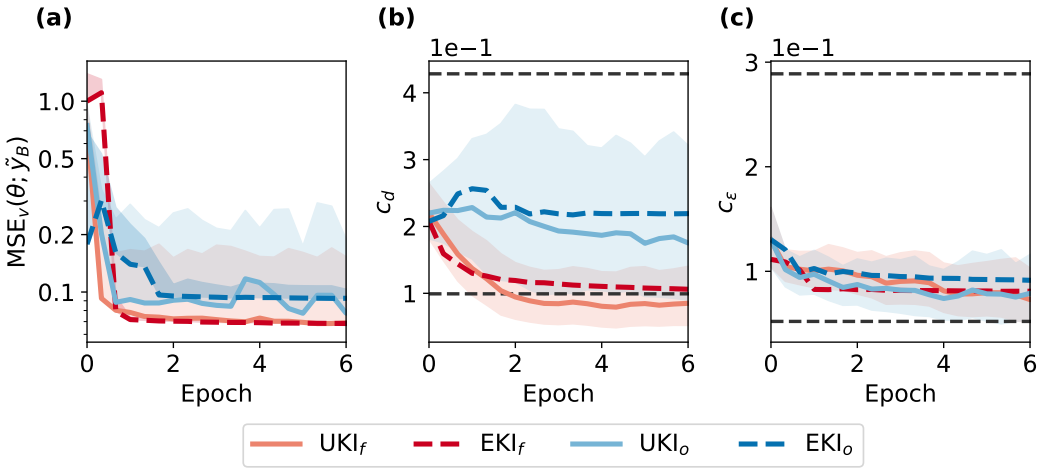

Figure 6: Evolution of the validation error (a) and estimates of the turbulent dissipation (b) and entrainment coefficient (c) for calibration processes using observations of the state (36) at 50 m resolution (UKI$_f$, EKI$_f$), or from $\bar{\theta}_l, \bar{q}_t$ and $\bar{q}_l$ at 200 m resolution (UKI$_o$, EKI$_o$). All inversion processes use $|B| = 20$. Shading is defined as in Figures 2 and 3.

for calibration, since it enables the use of ensembles with $J < 2p + 1$. In Figure 8, we present training and validation errors for the hybrid model using $J = 50, 100$ and 159, and for the empirical EDMF scheme with $J = 2p + 1 = 33$ ensemble members. We initialize the neural network weights as $\theta_{\mathrm{NN}} \sim \mathcal{N}(\theta^0_{\mathrm{NN}}, I)$ with $\theta^0_{\mathrm{NN}} \sim U(-0.05, 0.05)$. In all cases, we use L_2 regularization as discussed in Section 4.2 for all parameters except for those pertaining to entrainment and detrainment. We do this to showcase the regularization provided by the compact support property of EKI (Schillings & Stuart, 2017). We calibrate all parameters of the empirical and hybrid models, to compare the optimal performance of both closures.

Both the empirical and hybrid EDMF schemes generalize well to the validation set, with training and validation errors reaching levels of about 5% of the largest a priori validation error. The strong generalization to 4 K-warmer cloud regimes contrasts with results from approaches that try to learn unresolved tendencies directly, without encoding the physics (Rasp et al., 2018). Using a physics-based approach, all learned closures are consistently placed within the coarse-grained dynamics of the system (Cohen et al., 2020), which also vastly reduces the need to overparameterize unknown processes. Further, targeting closure terms which isolate a single physical process lends itself to interpretability in a manner difficult for purely machine-learning based parameterizations that simultaneously model many physical processes. After training, relationships between EDMF variables and targeted physical quantities like entrainment can be teased out using partial dependence plots or ablation studies. In addition, the learned relationships are pointwise and causal.

The inset in Figure 8b shows how the higher-complexity hybrid model moderately overfits to the training set after ~ 10 epochs, a behavior that is not observed with the empirical model. Hence, in the low-data regime ($d \lesssim p$), adoption of techniques such as early stopping (Prechelt, 1998) or sparsity-inducing regularization (Schneider et al., 2020) becomes necessary. The compact support property of EKI, which mandates that the solution be in the linear span of the initial ensemble, also regularizes the learned hybrid model with decreasing J; for $J = 50 < p$ overfitting is minimal. Thus, reducing the ensemble size is an efficient regularization technique when training large machine-

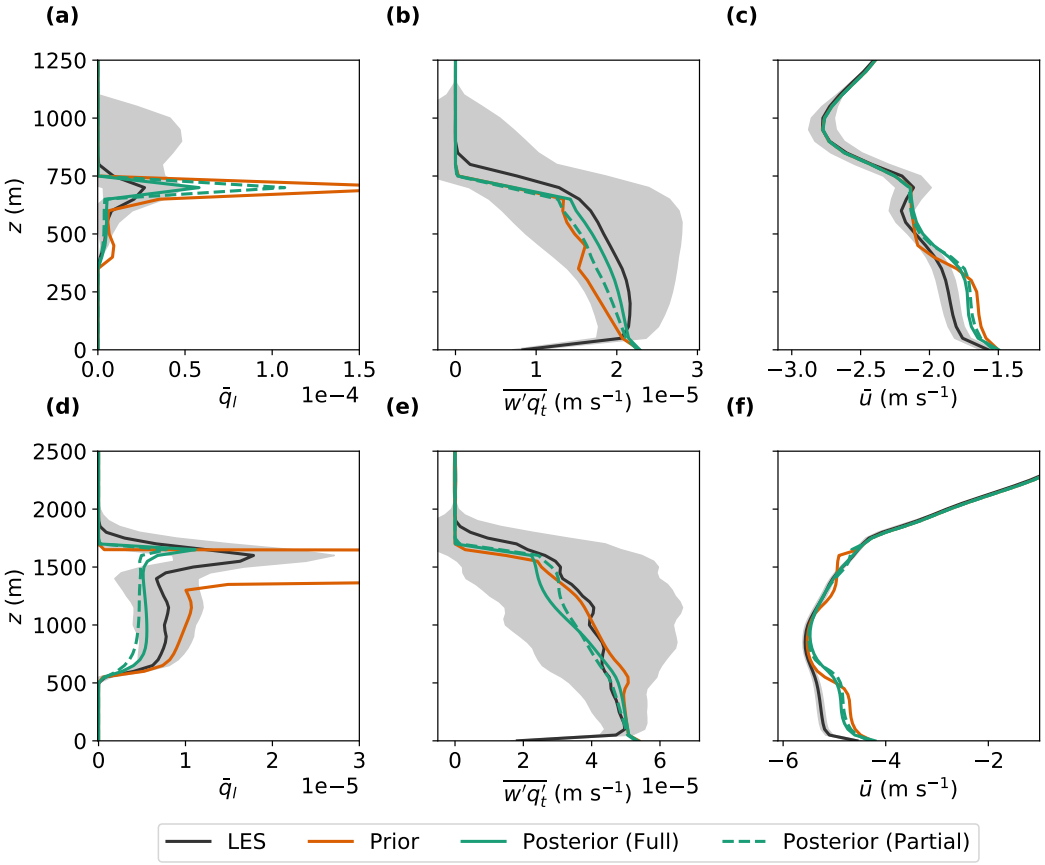

Figure 7: Prior, posterior and LES profiles of liquid water specific humidity ($\bar{q}_l$), vertical moisture flux ($\overline{w'q'_t}$) and zonal velocity ($\bar{u}$) for cfSite 3 using July forcing (top) and cfSite 14 using January forcing (bottom) from the AMIP4K experiment (Shen et al., 2022). Posterior results are shown for a model calibrated using the high-resolution state (36) (Full), and coarse-resolution observations of $\bar{\theta}_l, \bar{q}_t$ and $\bar{q}_l$ (Partial). Shading and legend as in Figure 5. Results obtained using UKI with $|B| = 20$.

learning models that tend to overfit. Additional EKI-specific regularization techniques for deeper networks are discussed in Kovachki and Stuart (2019).

Another difference between the empirical and the hybrid models is that for the latter, we do not know a priori the parameter ranges for which the model trajectories remain physical. During the training sessions shown in Figure 8, the hybrid models experienced a maximum of 25 ($J = 50$), 30 ($J = 100$) and 72 ($J = 159$) failures in a single iteration, all occurring during the first epoch. The use of the modified failsafe update proposed in Section 3.1.1 proved crucial to enable training in the presence of model failures, and it reduced the number of failures to a small fraction of J after a few EKI iterations.

Profiles of $\bar{q}_l, \bar{q}_t$ and $\overline{w'q'_t}$ are shown in Figure 9 for the trained empirical and hybrid EDMF models. To produce the profiles with the hybrid model, we retain the parameters learned at the iteration with lowest validation error from a training session spanning 25 epochs, effectively similar to early stopping. As expected from the validation error, the hybrid model slightly improves upon the skill of the empirical model, predict-

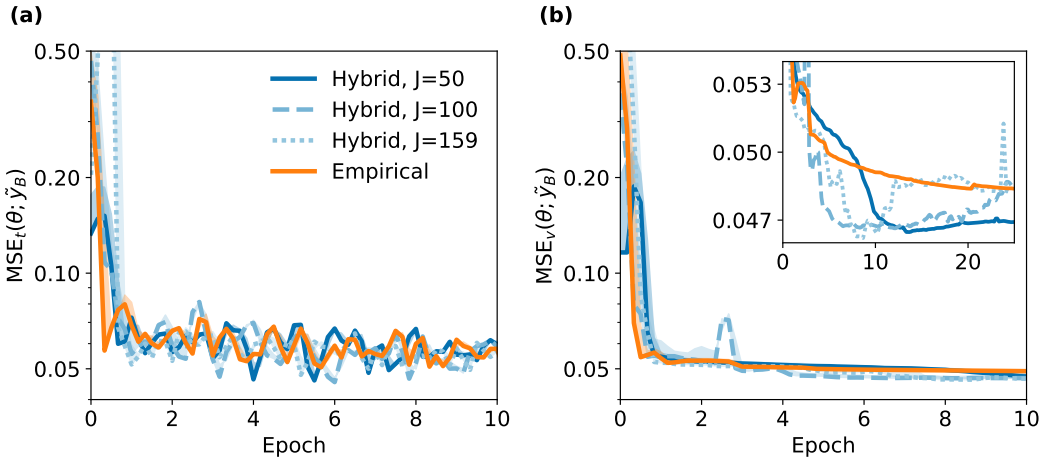

Figure 8: Batch (a) training and (b) validation normalized mean squared error for the hybrid and empirical EDMF models. Lines, shading and inset as in Figure 2. Results are shown for calibration with EKI, using $J = 50, 100$ and $2p + 1 = 159$ ensemble members for the hybrid model. The empirical model training uses $J = 2p + 1 = 33$. All inversion processes use batch size $|B| = 10$.

ing more accurate profiles of $\bar{q}_l$ within the cloud layer. This is, of course, at the cost of a significantly higher parameter complexity of the closure.

As shown here, ensemble Kalman inversion allows for rapid prototyping and comparison of closures within an overarching *black-box* model. Importantly, this comparison can be done in terms of the online performance of the fully calibrated overarching model.

5 Discussion and conclusions

Ensemble Kalman methods such as ensemble and unscented Kalman inversion are powerful tools for training possibly expensive models. They do not impose any constraint on the data used for learning, or the architecture of the closures to be calibrated. This means that ensemble Kalman methods can be used to learn all parameters within arbitrarily complex overarching models, regardless of where those parameters appear in the formulation of the model.

This enables training physics-based machine-learning parameterizations, as demonstrated here by substituting an internal component of the EDMF model by a neural network, which required no change in the data or framework used for training. The benefits of combining physics and data are evinced by the performance of our trained hybrid closure in simulations of clouds typical of conditions 4 K warmer than the clouds in the training set.

In order to use these algorithms, parameter learning must be framed as an inverse problem. This allows great flexibility, but raises the problem of choosing a reasonable observational map $\mathcal{H}$ and prior covariance Γ to define an inverse problem when we have access to high-dimensional data. Through a domain-agnostic strategy and a reasonable heuristic about the expected model error, we have demonstrated a systematic way of constructing a well-defined inverse problem from high-dimensional data. This strategy is designed to maximize the information content through a lossy principal component encoding $\mathcal{H}$ and to allow the use of time averages as observations, making it amenable to har-

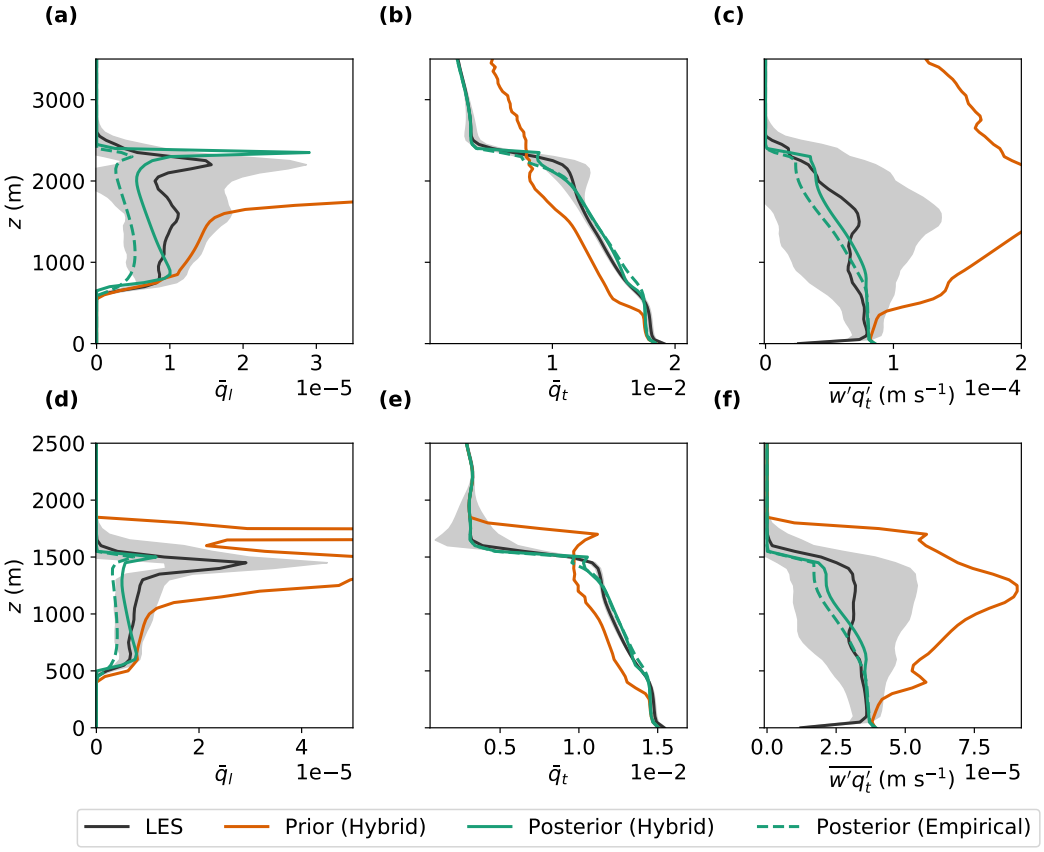

Figure 9: Prior, posterior and LES profiles of liquid water specific humidity ($\bar{q}_l$), total water specific humidity ($\bar{q}_t$) and vertical moisture flux ($\overline{w'q'_t}$) for cfSite 14 using July forcing (top) and cfSite 8 using January forcing (bottom) from the AMIP4K experiment (Shen et al., 2022). Definitions of prior, posterior and shading as in Figure 5. Posterior results are shown for the EDMF model with empirical closures (Empirical), and with the neural network entrainment closure (38) (Hybrid), using early stopping and 25 epochs of training. Results obtained using EKI with $|B| = 10$.

nessing, e.g., satellite observations in addition to computationally generated data. The success of this strategy is demonstrated in a variety of settings, using empirical and hybrid models.

Nevertheless, the flexibility of the inverse problem allows to define the observational map $\mathcal{H}$ through any observable diagnostic of the model, be it differentiable or not. For instance, Barthélémy et al. (2021) use a neural network as the mapping $\mathcal{H}$, to train a low-resolution dynamical model directly from features at high resolution. One could also envision the construction of $\mathcal{H}$ through other statistics of the model dynamics, such as the variance or skewness. These choices may be preferable for particular tasks, such as the prediction of extreme events or the correct representation of emergent phenomena.

Given an inverse problem, we have shown that EKI and UKI are robust to noise and amenable to batching strategies. This establishes the ability of the Kalman algorithms to train models using sequentially sampled data. The same robustness can be expected for other sources of noise, such as stochasticity in the model, as shown by Schneider et al. (2021). In addition, we have proposed modifications of the EKI and UKI updates

that enable calibrating models that may fail during training, which is often the case for Earth system models.

Although similar, each inversion algorithm presents its own relative strengths in our analysis. Calibration through EKI appears to be more robust to noise, and the number of ensemble members may be chosen to be lower than for UKI when the parameter space is high-dimensional. Indeed, Kovachki and Stuart (2019) show successful results for EKI when the number of parameters (e.g., $p \sim 10^6$) is two orders of magnitude higher than the ensemble size. Using fewer ensemble members than parameters also introduces a regularization effect. On the other hand, UKI provides information about parametric uncertainty and correlations, which can be used to improve models at the process level, and to rapidly compare the added value of increasingly precise observing systems. Other ensemble Kalman methods, such as the sparsity-inducing EKI (Schneider et al., 2020) or the ensemble Kalman sampler (Garbuno-Inigo et al., 2020), can provide solutions to the inverse problem with other useful properties. In addition, all these ensemble methods generate parameter-output pairs that can be used to train emulators for uncertainty quantification (Cleary et al., 2021).

Finally, ensemble Kalman methods may be used for the rapid comparison of parameterizations in terms of the online skill of an overarching Earth system model. The same framework could be used to train Gaussian processes, random feature models (Nelsen & Stuart, 2020), Fourier neural operators (Z. Li et al., 2020), or stochastic closures (Guillaumin & Zanna, 2021), for example. These are some of the exciting research avenues that we will be exploring in the future.

Appendix A Configuration-based principal component analysis

Performing PCA on each configuration allows retaining principal modes from low-variance configurations while filtering out trailing modes from high-variance configurations. The importance of this is demonstrated in Figure A1 for three configurations of our LES solver (Pressel et al., 2015) based on observational campaigns of a stable boundary layer, a stratocumulus-topped boundary layer, and shallow cumulus convection (Beare et al., 2006; Stevens et al., 2005; Siebesma et al., 2003). Performing global PCA is equivalent to using a cutoff $\sigma^2 > \sigma_*^2$ in Figure A1a, where we need to choose between neglecting most modes from certain configurations (e.g., GABLS in Figure A1a) or retaining highly oscillatory modes from others (e.g., Bomex), as measured by the number of zero-crossings of the eigenmode (Hansen, 1998). Highly oscillatory modes may have a disproportionate contribution to the loss when calibrating imperfect models. On the other hand, performing PCA on each $\tilde{\Gamma}_c$ alleviates this problem by aligning the eigenspectra before applying the cutoff, as shown in Figure A1b. Appropriate conditioning of the global covariance matrix is still enforced when applying configuration-based PCA through the Tikhonov regularizer in equation (14).

Appendix B Addressing model failures with unscented Kalman inversion

In the presence of model failures, we perform the UKI quadratures over the successful sigma points. Consider the set of off-center sigma points $\{\hat{\theta}\} = \{\hat{\theta}_s\} \cup \{\hat{\theta}_f\}$ where $\hat{\theta}_s^{(j)}$, $j = 1, \dots, J_s$ are successful members and $\hat{\theta}_f^{(k)}$ are not. For ease of notation, consider an ordering of $\{\hat{\theta}\}$ such that $\{\hat{\theta}_s\}$ are its first J_s elements, and note that we deal with the central point $\hat{\theta}^{(0)}$ separately. We estimate the covariances $\mathrm{Cov}_q(\mathcal{G}_n, \mathcal{G}_n)$ and $\mathrm{Cov}_q(\theta_n, \mathcal{G}_n)$ from the successful ensemble,

$$\mathrm{Cov}_q(\theta_n, \mathcal{G}_n) \approx \sum_{j=1}^{J_s} w_{s,j} (\hat{\theta}_{s,n}^{(j)} - \bar{\theta}_{s,n})(\mathcal{G}(\hat{\theta}_{s,n}^{(j)}) - \bar{\mathcal{G}}_{s,n})^T, \tag{B1}$$

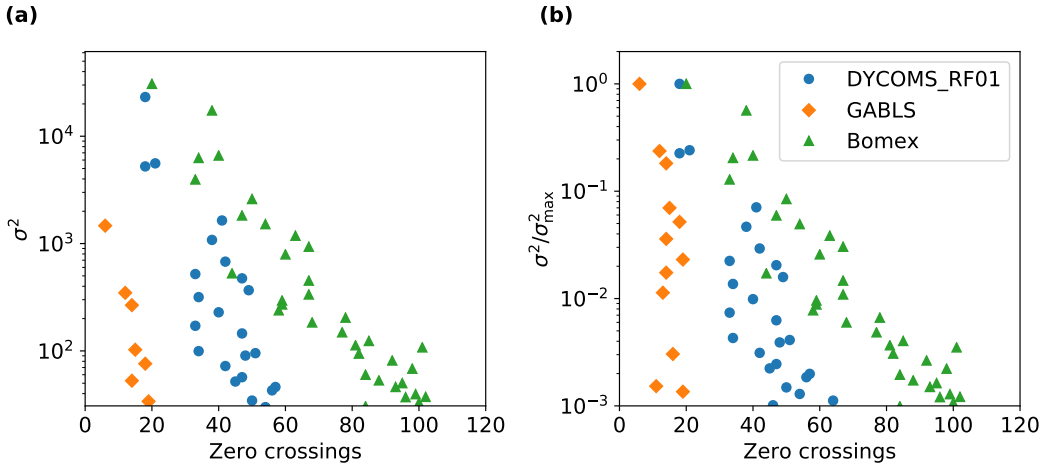

Figure A1: (a) Scatter plot of covariance eigenvalues σ^2 and the number of zero-crossings of their corresponding eigenmode for three different configurations of an LES solver. (b) The same plot, with eigenvalues normalized by the leading eigenvalue of each configuration ($\sigma^2_{\max}$). Trailing eigenvalues are associated with high-wavenumber oscillatory modes with frequent sign changes.

$$\mathrm{Cov}_q(\mathcal{G}_n, \mathcal{G}_n) \approx \sum_{j=1}^{J_s} w_{s,j} (\mathcal{G}(\hat{\theta}^{(j)}_{s,n}) - \bar{\mathcal{G}}_{s,n})(\mathcal{G}(\hat{\theta}^{(j)}_{s,n}) - \bar{\mathcal{G}}_{s,n})^T, \tag{B2}$$

where the weights at each successful sigma point are scaled up, to preserve the sum of weights,

$$w_{s,j} = \left(\frac{\sum_{i=1}^{2p} w_i}{\sum_{k=1}^{J_s} w_k} \right) w_j. \tag{B3}$$

In equations (B1) and (B2), the means $\bar{\theta}_{s,n}$ and $\bar{\mathcal{G}}_{s,n}$ must be modified from the original formulation if the central point $\hat{\theta}^{(0)} = m_n$ results in model failure,

$$\bar{\theta}_{s,n} = \begin{cases} m_n & \text{if successful,} \\ \dfrac{1}{J_s} \sum_{j=1}^{J_s} \hat{\theta}^{(j)}_{s,n} & \text{otherwise,} \end{cases} \tag{B4}$$

$$\bar{\mathcal{G}}_{s,n} = \begin{cases} \mathcal{G}(m_n) & \text{if successful,} \\ \dfrac{1}{J_s} \sum_{j=1}^{J_s} \mathcal{G}(\hat{\theta}^{(j)}_{s,n}) & \text{otherwise.} \end{cases} \tag{B5}$$

These modified UKI quadrature rules are used throughout Section 4 to deal with model failures. Since UKI can be initialized from a tighter prior than EKI, due to the absence of ensemble collapse, failures are much easier to avoid than with EKI.

Appendix C Parameter transformation and prior

Given a prior range $[\phi_i, \phi_f]$ for a parameter $\phi \in \mathbb{R}$, we define the transformation

$$\theta = \mathcal{T}(\phi) = \ln \frac{\phi - \phi_i}{\phi_f - \phi}, \tag{C1}$$

such that the interval midpoint is mapped to $\theta = 0$, and the bounds to $\pm\infty$. An unconstrained Gaussian prior may then be defined for θ given the prior mean in physical

(constrained) parameter space ϕ_p as

$$\theta_0 \sim \mathcal{N}(\mathcal{T}(\phi_p), \sigma_0^2), \tag{C2}$$

where σ_0^2 is a free parameter controlling the size of the region within the interval $[\phi_i, \phi_f]$ containing most of the probability. This means that the magnitude of σ is already normalized with respect to the prior range, so we will generally choose $\sigma \sim \mathcal{O}(1)$. The $p-$dimensional prior $\mathcal{N}(m_0, \Sigma_0)$ is then constructed as an uncorrelated multivariate normal with marginal distributions given by expression (C2). The normalization induced by (C1) also enables the use of isotropic regularization in equation (35), even though the physical parameters ϕ may differ in order of magnitude. For more examples of parameter transformations in the context of EKI and UKI, see Huang et al. (2022) and Dunbar et al. (2022).

Appendix D Calibration using very noisy loss evaluations

The Kalman inversion results are expected to deteriorate above some noise threshold, as the signal-to-noise ratio in the training process decreases. We explored the sensitivity of UKI and EKI to noise by sampling a single configuration per iteration from the training set described in Section 4.2. As shown in Figure D1, UKI fails to converge to the minimum found with larger batches in this limit. The validation error is characterized by large oscillations due to strong changes in the value of model parameters like the entrainment coefficient c_ϵ or the eddy diffusivity coefficient c_m. On the other hand, EKI proves robust to noise even in this limit, converging to the minimum found by UKI employing larger batches.

In the context of Kalman inversion, decreasing the step size Δt is equivalent to increasing the noise variance, as shown in updates (20) and (27). We investigate the time step role in the small batch limit by performing the ensemble Kalman inversion with $\Delta t = |C|^{-1} = 1/60$. The smaller time step increases the parameter uncertainty, which leads to a reduction in parameter oscillations and estimates closer to the prior. This is accompanied by a moderate reduction in validation error oscillations. We performed additional inversions using even smaller time steps, which resulted in a convergence of the parameter estimates towards the prior and a minor reduction in validation error with respect to the initialization. We conclude that decreasing Δt in UKI can reduce oscillations due to high levels of noise, but it does not result in the same robustness as EKI.

Acknowledgments

We thank Daniel Z. Huang and Zhaoyi Shen for insightful discussions. I.L. was supported by a fellowship from the Resnick Sustaintability Institute at Caltech, and an Amazon AI4Science fellowship. This research was additionally supported by the generosity of Eric and Wendy Schmidt by recommendation of the Schmidt Futures program, by the National Science Foundation (grant AGS-1835860), and by the Heising-Simons Foundation. Part of this research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. The software package implementing ensemble Kalman methods can be accessed at `https://doi.org/10.5281/zenodo.6382968`, and the software used to calibrate the EDMF scheme may be accessed at `https://doi.org/10.5281/zenodo.6382865`. The data from Shen et al. (2022) used for model training is available at `https://doi.org/10.22002/D1.20052`.

References

Barthélémy, S., Brajard, J., Bertino, L., & Counillon, F. (2021). *Super-resolution data assimilation.* doi: 10.48550/arxiv.2109.08017

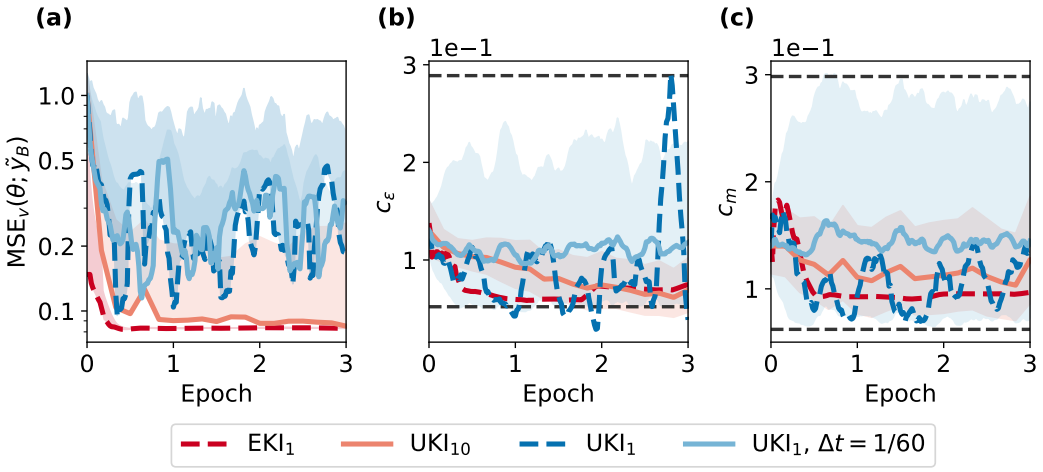

Figure D1: Evolution of the validation error (a) and estimates of the entrainment (b), and eddy diffusivity (c) coefficients. Results shown for UKI using batch sizes of 10 and 1, and EKI using a batch size of 1. Parameter uncertainty only shown for UKI_{10} and UKI_1, $\Delta t = 1/60$ for clarity. All results shown use $\Delta t = |C|/|B|$ unless otherwise specified. Shading as in Figures 2 and 3.

Beare, R. J., Macvean, M. K., Holtslag, A. A. M., Cuxart, J., Esau, I., Golaz, J.-C., ... Sullivan, P. (2006). An intercomparison of large-eddy simulations of the stable boundary layer. *Boundary-Layer Meteorology*, *118*, 247–272. doi: 10.1007/s10546-004-2820-6

Beucler, T., Pritchard, M., Rasp, S., Ott, J., Baldi, P., & Gentine, P. (2021). Enforcing Analytic Constraints in Neural Networks Emulating Physical Systems. *Physical Review Letters*, *126*, 98302. doi: 10.1103/PhysRevLett.126.098302

Brajard, J., Carrassi, A., Bocquet, M., & Bertino, L. (2021). Combining data assimilation and machine learning to infer unresolved scale parametrization. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *379*, 20200086. doi: 10.1098/rsta.2020.0086

Brenowitz, N. D., Beucler, T., Pritchard, M., & Bretherton, C. S. (2020). Interpreting and stabilizing machine-learning parametrizations of convection. *Journal of the Atmospheric Sciences*, *77*, 4357-4375. doi: 10.1175/JAS-D-20-0082.1

Brenowitz, N. D., & Bretherton, C. S. (2018). Prognostic validation of a neural network unified physics parameterization. *Geophysical Research Letters*, *45*, 6289-6298. doi: 10.1029/2018GL078510

Bretherton, C. S., Henn, B., Kwa, A., Brenowitz, N. D., Watt-Meyer, O., McGibbon, J., ... Harris, L. (2022). Correcting coarse-grid weather and climate models by machine learning from global storm-resolving simulations. *Journal of Advances in Modeling Earth Systems*, *14*, e2021MS002794. doi: 10.1029/2021MS002794

Chada, N. K., Stuart, A. M., & Tong, X. T. (2020). Tikhonov regularization within ensemble Kalman inversion. *SIAM Journal on Numerical Analysis*, *58*, 1263-1294. doi: 10.1137/19M1242331

Cleary, E., Garbuno-Inigo, A., Lan, S., Schneider, T., & Stuart, A. M. (2021). Calibrate, emulate, sample. *Journal of Computational Physics*, *424*, 109716. doi: 10.1016/j.jcp.2020.109716

Cohen, Y., Lopez-Gomez, I., Jaruga, A., He, J., Kaul, C. M., & Schneider, T. (2020). Unified entrainment and detrainment closures for extended eddy-diffusivity mass-flux schemes. *Journal of Advances in Modeling Earth Systems*,

12, e2020MS002162. doi: 10.1029/2020MS002162

Couvreux, F., Hourdin, F., Williamson, D., Roehrig, R., Volodina, V., Villefranque, N., ... Xu, W. (2021). Process-based climate model development harnessing machine learning: I. a calibration tool for parameterization improvement. *Journal of Advances in Modeling Earth Systems*, *13*, e2020MS002217. doi: 10.1029/2020MS002217

de Roode, S. R., Duynkerke, P. G., & Siebesma, A. P. (2000). Analogies between mass-flux and reynolds-averaged equations. *Journal of the Atmospheric Sciences*, *57*, 1585-1598. doi: 10.1175/1520-0469(2000)057⟨1585:ABMFAR⟩2.0.CO;2

Dunbar, O., Howland, M. F., Schneider, T., & Stuart, A. (2022). Ensemble-based experimental design for targeted high-resolution simulations to inform climate models. *Earth and Space Science Open Archive*, 24. doi: 10.1002/essoar.10510142.1

Garbuno-Inigo, A., Hoffmann, F., Li, W., & Stuart, A. M. (2020). Interacting Langevin diffusions: Gradient structure and ensemble Kalman sampler. *SIAM Journal on Applied Dynamical Systems*, *19*, 412-441. doi: 10.1137/19M1251655

Guillaumin, A. P., & Zanna, L. (2021). Stochastic-deep learning parameterization of ocean momentum forcing. *Journal of Advances in Modeling Earth Systems*, *13*, e2021MS002534. doi: 10.1029/2021MS002534

Hansen, P. C. (1990). Truncated singular value decomposition solutions to discrete ill-posed problems with ill-determined numerical rank. *SIAM Journal on Scientific and Statistical Computing*, *11*, 503-518. doi: 10.1137/0911028

Hansen, P. C. (1998). *Rank-deficient and discrete ill-posed problems.* Society for Industrial and Applied Mathematics. doi: 10.1137/1.9780898719697

He, J., Cohen, Y., Lopez-Gomez, I., Jaruga, A., & Schneider, T. (2021). An improved perturbation pressure closure for eddy-diffusivity mass-flux schemes. *Earth and Space Science Open Archive*, 37. doi: 10.1002/essoar.10505084.2

Huang, D. Z., Schneider, T., & Stuart, A. M. (2022). *Iterated Kalman methodology for inverse problems.* doi: 10.48550/arxiv.2102.01580

Iglesias, M. A., Law, K. J. H., & Stuart, A. M. (2013). Ensemble Kalman methods for inverse problems. *Inverse Problems*, *29*, 045001. doi: 10.1088/0266-5611/29/4/045001

Kennedy, M., & O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society Series B*, *63*, 425-464. doi: 10.1111/1467-9868.00294

Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., & Tang, P. T. P. (2016). *On large-batch training for deep learning: Generalization gap and sharp minima.* doi: 10.48550/arXiv.1609.04836

Kovachki, N. B., & Stuart, A. M. (2019). Ensemble Kalman inversion: a derivative-free technique for machine learning tasks. *Inverse Problems*, *35*, 095005. doi: 10.1088/1361-6420/ab1c3a

Li, M., Zhang, T., Chen, Y., & Smola, A. J. (2014). Efficient mini-batch training for stochastic optimization. ACM. doi: 10.1145/2623330.2623612

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., & Anandkumar, A. (2020). *Fourier neural operator for parametric partial differential equations.* doi: 10.48550/arxiv.2010.08895

Ling, J., Kurzawski, A., & Templeton, J. (2016). Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *Journal of Fluid Mechanics*, *807*, 155-166. doi: 10.1017/jfm.2016.615

Lopez-Gomez, I., Cohen, Y., He, J., Jaruga, A., & Schneider, T. (2020). A generalized mixing length closure for eddy-diffusivity mass-flux schemes of turbulence and convection. *Journal of Advances in Modeling Earth Systems*, *12*, e2020MS002161. doi: 10.1029/2020MS002161

Meyer, D., Hogan, R. J., Dueben, P. D., & Mason, S. L. (2022). Machine learning emulation of 3D cloud radiative effects. *Journal of Advances in Modeling Earth Systems*, *14*, e2021MS002550. doi: 10.1029/2021MS002550

Mlawer, E. J., Taubman, S. J., Brown, P. D., Iacono, M. J., & Clough, S. A. (1997). Radiative transfer for inhomogeneous atmospheres: RRTM, a validated correlated-k model for the longwave. *Journal of Geophysical Research: Atmospheres*, *102*, 16663–16682. doi: 10.1029/97JD00237

Morzfeld, M., Adams, J., Lunderman, S., & Orozco, R. (2018). Feature-based data assimilation in geophysics. *Nonlinear Processes in Geophysics*, *25*, 355-374. doi: 10.5194/npg-25-355-2018

Muñoz-Sabater, J., Dutra, E., Agustí-Panareda, A., Albergel, C., Arduini, G., Balsamo, G., ... Thépaut, J.-N. (2021). ERA5-Land: a state-of-the-art global reanalysis dataset for land applications. *Earth System Science Data*, *13*, 4349-4383. doi: 10.5194/essd-13-4349-2021

National Academies of Sciences, Engineering, and Medicine. (2018). *Thriving on our changing planet: A decadal strategy for earth observation from space.* Washington, DC: The National Academies Press. doi: 10.17226/24938

Nelsen, N. H., & Stuart, A. M. (2020). *The random feature model for input-output maps between banach spaces.* doi: 10.48550/arxiv.2005.10224

Pathak, J., Subramanian, S., Harrington, P., Raja, S., Chattopadhyay, A., Mardani, M., ... Anandkumar, A. (2022). FourCastNet: A global data-driven high-resolution weather model using adaptive fourier neural operators. doi: 10.48550/arxiv.2202.11214

Prechelt, L. (1998). *Early stopping - but when?* Springer Berlin Heidelberg. doi: 10.1007/3-540-49430-8_3

Pressel, K. G., Kaul, C. M., Schneider, T., Tan, Z., & Mishra, S. (2015). Large-eddy simulation in an anelastic framework with closed water and entropy balances. *Journal of Advances in Modeling Earth Systems*, *7*, 1425-1456. doi: 10.1002/2015MS000496

Rasp, S., Pritchard, M. S., & Gentine, P. (2018). Deep learning to represent subgrid processes in climate models. *Proceedings of the National Academy of Sciences*, *115*, 9684-9689. doi: 10.1073/pnas.1810286115

Rasp, S., & Thuerey, N. (2021). Data-driven medium-range weather prediction with a Resnet pretrained on climate simulations: A new model for WeatherBench. *Journal of Advances in Modeling Earth Systems*, *13*, e2020MS002405. doi: 10.1029/2020MS002405

Ravuri, S., Lenc, K., Willson, M., Kangin, D., Lam, R., Mirowski, P., ... Mohamed, S. (2021). Skilful precipitation nowcasting using deep generative models of radar. *Nature*, *597*, 672-677. doi: 10.1038/s41586-021-03854-z

Reichel, L., & Rodriguez, G. (2013). Old and new parameter choice rules for discrete ill-posed problems. *Numerical Algorithms*, *63*, 65-87.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., & Prabhat. (2019). Deep learning and process understanding for data-driven earth system science. *Nature*, *566*, 195–204.

Schillings, C., & Stuart, A. M. (2017). Analysis of the ensemble Kalman filter for inverse problems. *SIAM Journal on Numerical Analysis*, *55*, 1264-1290. doi: 10.1137/16M105959X

Schmit, T. J., Griffith, P., Gunshor, M. M., Daniels, J. M., Goodman, S. J., & Lebair, W. J. (2017). A closer look at the ABI on the GOES-R series. *Bulletin of the American Meteorological Society*, *98*, 681-698. doi: 10.1175/BAMS-D-15-00230.1

Schneider, T., Lan, S., Stuart, A., & Teixeira, J. (2017). Earth system modeling 2.0: A blueprint for models that learn from observations and targeted high-resolution simulations. *Geophysical Research Letters*, *44*, 312-396. doi: 10.1002/2017GL076101

Schneider, T., Stuart, A. M., & Wu, J.-L. (2020). *Ensemble Kalman inversion for sparse learning of dynamical systems from time-averaged data.* doi: 10.48550/arxiv.2007.06175

Schneider, T., Stuart, A. M., & Wu, J.-L. (2021). Learning stochastic closures using ensemble Kalman inversion. *Transactions of Mathematics and Its Applications*, *5*, tnab003. doi: 10.1093/imatrm/tnab003

Seifert, A., & Rasp, S. (2020). Potential and limitations of machine learning for modeling warm-rain cloud microphysical processes. *Journal of Advances in Modeling Earth Systems*, *12*, e2020MS002301. doi: 10.1029/2020MS002301

Shen, Z., Sridhar, A., Tan, Z., Jaruga, A., & Schneider, T. (2022). A library of large-eddy simulations forced by global climate models. *Journal of Advances in Modeling Earth Systems*, e2021MS002631. doi: 10.1029/2021MS002631

Siebesma, A. P. (1996). On the mass flux approach for atmospheric convection. In *Workshop on new insights and approaches to convective parametrization, 4-7 november 1996* (p. 25-57). Shinfield Park, Reading: ECMWF.

Siebesma, A. P., Bretherton, C. S., Brown, A., Chlond, A., Cuxart, J., Duynkerke, P. G., ... others (2003). A large eddy simulation intercomparison study of shallow cumulus convection. *Journal of the Atmospheric Sciences*, *60*, 1201–1219.

Stevens, B., Moeng, C.-H., Ackerman, A. S., Bretherton, C. S., Chlond, A., de Roode, S., ... Zhu, P. (2005). Evaluation of large-eddy simulations via observations of nocturnal marine stratocumulus. *Monthly Weather Review*, *133*, 1443-1462. doi: 10.1175/MWR2930.1

Suselj, K., Posselt, D., Smalley, M., Lebsock, M. D., & Teixeira, J. (2020). A New Methodology for Observation-Based Parameterization Development. *Monthly Weather Review*, *148*, 4159–4184. doi: 10.1175/MWR-D-20-0114.1

Sønderby, C. K., Espeholt, L., Heek, J., Dehghani, M., Oliver, A., Salimans, T., ... Kalchbrenner, N. (2020). *Metnet: A neural weather model for precipitation forecasting.* doi: 10.48550/arXiv.2003.12140

Tan, Z., Kaul, C. M., Pressel, K. G., Cohen, Y., Schneider, T., & Teixeira, J. (2018). An extended eddy-diffusivity mass-flux scheme for unified representation of subgrid-scale turbulence and convection. *Journal of Advances in Modeling Earth Systems*, *10*, 770-800. doi: 10.1002/2017MS001162

Tong, X. T., & Morzfeld, M. (2022). *Localization in ensemble Kalman inversion.* doi: 10.48550/arXiv.2201.10821

Villefranque, N., Blanco, S., Couvreux, F., Fournier, R., Gautrais, J., Hogan, R. J., ... Williamson, D. (2021). Process-based climate model development harnessing machine learning: III. The representation of cumulus geometry and their 3D radiative effects. *Journal of Advances in Modeling Earth Systems*, *13*, e2020MS002423. doi: 10.1029/2020MS002423

Weyn, J. A., Durran, D. R., Caruana, R., & Cresswell-Clay, N. (2021). Sub-seasonal forecasting with a large ensemble of deep-learning weather prediction models. *Journal of Advances in Modeling Earth Systems*, *13*, e2021MS002502. doi: 10.1029/2021MS002502

Zanna, L., & Bolton, T. (2020). Data-driven equation discovery of ocean mesoscale closures. *Geophysical Research Letters*, *47*, e2020GL088376. doi: 10.1029/2020GL088376

Zhao, W. L., Gentine, P., Reichstein, M., Zhang, Y., Zhou, S., Wen, Y., ... Qiu, G. Y. (2019). Physics-constrained machine learning of evapotranspiration. *Geophysical Research Letters*, *46*, 14496-14507. doi: 10.1029/2019GL085291