AGU ADVANCING EARTH AND SPACE SCIENCES

JAMES | Journal of Advances in Modeling Earth Systems®

RESEARCH ARTICLE
10.1029/2023MS004178

Special Collection:
Data assimilation for Earth system models

Data Assimilation in Chaotic Systems Using Deep Reinforcement Learning

Mohamad Abed El Rahman Hammoud[1], Naila Raboudi[1], Edriss S. Titi[2,3], Omar Knio[1], and Ibrahim Hoteit[1]

[1]King Abdullah University of Science and Technology, Thuwal, Saudi Arabia, [2]Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge, UK, [3]Department of Mathematics, Texas A & M University, College Station, TX, USA

Key Points:

- Deep reinforcement learning (RL) is introduced for data assimilation in application to the Lorenz 63 and 96
- RL generalizes to new situations unseen during training through actively learning from the data and system dynamics
- The RL agent allows for nonlinear correction of the forecast using the observations
- The performance of the proposed RL algorithm generally surpasses that of the standard ensemble Kalman filter (EnKF)

Supporting Information:

Supporting Information may be found in the online version of this article.

Correspondence to:

I. Hoteit,
ibrahim.hoteit@kaust.edu.sa

Citation:

Hammoud, M. A. E. R., Raboudi, N., Titi, E. S., Knio, O., & Hoteit, I. (2024). Data assimilation in chaotic systems using deep reinforcement learning. *Journal of Advances in Modeling Earth Systems*, *16*, e2023MS004178. https://doi.org/10.1029/2023MS004178

Received 18 DEC 2023
Accepted 10 JUL 2024

Author Contributions:

Conceptualization: Mohamad Abed El Rahman Hammoud, Naila Raboudi, Edriss S. Titi, Omar Knio, Ibrahim Hoteit
Data curation: Mohamad Abed El Rahman Hammoud

Abstract Data assimilation (DA) plays a pivotal role in diverse applications, ranging from weather forecasting to trajectory planning for autonomous vehicles. A prime example is the widely used ensemble Kalman filter (EnKF), which relies on the Kalman filter's linear update equation to correct each of the ensemble forecast member's state with incoming observations. Recent advancements have witnessed the emergence of deep learning approaches in this domain, primarily within a supervised learning framework. However, the adaptability of such models to untrained scenarios remains a challenge. In this study, we introduce a new DA strategy that utilizes reinforcement learning (RL) to apply state corrections using full or partial observations of the state variables. Our investigation focuses on demonstrating this approach to the chaotic Lorenz 63 and 96 systems, where the agent's objective is to maximize the geometric series with terms that are proportional to the negative root-mean-squared error (RMSE) between the observations and corresponding forecast states. Consequently, the agent develops a correction strategy, enhancing model forecasts based on available observations. Our strategy employs a stochastic action policy, enabling a Monte Carlo-based DA framework that relies on randomly sampling the policy to generate an ensemble of assimilated realizations. Numerical results demonstrate that the developed RL algorithm performs favorably when compared to the EnKF. Additionally, we illustrate the agent's capability to assimilate non-Gaussian observations, addressing one of the limitations of the EnKF.

Plain Language Summary Reliable forecasts of the state of chaotic systems, such as environmental flows, require combining observational data and dynamical model outputs through a process called data assimilation. The ensemble Kalman filter (EnKF) is the most commonly adopted algorithm for this task, however, is subject to some limitations when applied to nonlinear/non-Gaussian systems. Recently, there has been interest in using deep learning (DL), particularly within a supervised learning setup, for DA. However, making DL models work well in new situations that differ from those experienced during training is challenging. In this work, we propose a new DA approach that leverages reinforcement learning (RL). RL helps the system make corrections to its predictions based on incoming observations, even when the model has not been trained for those specific scenarios. Compared to the EnKF framework, RL offers a novel algorithm for nonlinear corrections of the forecasts. Numerical results show that the proposed RL algorithm outperforms the EnKF and demonstrates the RL agent's ability at addressing some shortcomings of the EnKF.

1. Introduction

Assimilating observations is essential for improving predictability of chaotic and dynamic physical systems. Chaotic dynamical systems, such as those describing climate and weather, involve inherent imperfections and extreme sensitivity to initial conditions, whereas the available observational data often carry significant uncertainties (Eckmann & Ruelle, 1985). To address the associated challenges, data assimilation (DA) combines real-world observations with numerical model outputs, continually refining model predictions by aligning them with newly acquired observations to enhance the accuracy and reliability of the predictions (Ott et al., 2004). DA techniques are broadly categorized as variational and sequential methods (Ghil & Malanotte-Rizzoli, 1991; Hoteit et al., 2018; Le Dimet & Talagrand, 1986; Lorenc, 2003). The ensemble Kalman filter (EnKF) represents one of the most popular sequential DA techniques, especially in the context of large-scale nonlinear systems (Evensen, 2003). Operating within a Bayesian probabilistic framework, the EnKF sequentially splits the state estimation process into cycles that alternate between forecast steps, driven by the system's dynamical model, and

Formal analysis: Mohamad Abed El Rahman Hammoud, Naila Raboudi, Edriss S. Titi, Omar Knio, Ibrahim Hoteit
Funding acquisition: Omar Knio, Ibrahim Hoteit
Investigation: Mohamad Abed El Rahman Hammoud
Methodology: Mohamad Abed El Rahman Hammoud, Naila Raboudi, Edriss S. Titi, Omar Knio, Ibrahim Hoteit
Project administration: Edriss S. Titi, Omar Knio, Ibrahim Hoteit
Resources: Omar Knio, Ibrahim Hoteit
Software: Mohamad Abed El Rahman Hammoud
Supervision: Edriss S. Titi, Omar Knio, Ibrahim Hoteit
Validation: Mohamad Abed El Rahman Hammoud, Naila Raboudi, Edriss S. Titi, Omar Knio, Ibrahim Hoteit
Visualization: Mohamad Abed El Rahman Hammoud
Writing – original draft: Mohamad Abed El Rahman Hammoud, Naila Raboudi
Writing – review & editing: Mohamad Abed El Rahman Hammoud, Naila Raboudi, Edriss S. Titi, Omar Knio, Ibrahim Hoteit

analysis steps, which updates the forecast with incoming data (Evensen, 2003). This approach enables Monte Carlo (MC) approximations of both state forecast and analysis distributions through an ensemble of state samples (Hoteit et al., 2008).

EnKF schemes are most commonly adopted when assimilating uncertain observations of the system states across diverse fields due to their robustness, capacity to handle complex and high-dimensional systems, and computational efficiency (Houtekamer & Mitchell, 1998). However, their applicability is not without constraints, particularly when the underlying assumptions are compromised. In particular, challenges may arise from the EnKF's reliance on the linear update equations of the Kalman filter, and the necessity for maintaining a Gaussian distribution within the ensemble, both of which become challenging in the presence of strong nonlinearities (Hoteit et al., 2008; Kalnay, 2002). Additionally, whereas the Gaussian assumption for both model and observational noise offers convenience, it may not hold in real-world scenarios, thereby limiting EnKF's performance, especially with strongly nonlinear models (Subramanian et al., 2012). In such cases, it is necessary to explore alternative approaches that are better suited for these scenarios; for example, van Leeuwen (2009).

Reinforcement Learning (RL) is a paradigm of artificial intelligence that deals with how an agent can learn to make decisions through interactions with an environment, namely to achieve a specific objective (Recht, 2019). It is inspired by behavioral psychology and focuses on learning how to take actions in an environment to maximize some notion of cumulative reward. Within the RL framework, an agent engages in trial-and-error exploration, testing various actions and observing their outcomes (Mnih et al., 2015). The agent's goal is to formulate an optimal strategy, often referred to as a policy, that guides its actions to maximize the cumulative reward over a time horizon. It is noteworthy to point out that the RL framework is inherently different from the supervised learning approaches because the latter require a pre-computed reference database for training, which in this context consists in minimizing a global objective function (Glorot & Bengio, 2010; Hammoud et al., 2023; Karniadakis et al., 2021). RL finds extensive applications in domains necessitating dynamic control and decision-making capabilities, encompassing fields such as robotics (Kober et al., 2013), gaming (Mnih et al., 2013; Vinyals et al., 2019), autonomous navigation (Sallab et al., 2017), fluid dynamics (Bae & Koumoutsakos, 2022; Novati et al., 2021), and beyond.

In this work, we introduce a new DA formalism utilizing RL as a strategy to actively update a nonlinear forecast correction scheme with the incoming data. The RL agent learns through interactions with the environment, adapting to its changes, and actively applies nonlinear corrections to handle complex processes. Numerical experiments were conducted with the Lorenz 63 (Lorenz, 1963) and 96 (Lorenz, 1996) chaotic systems, and the RL agent's performance was benchmarked against the EnKF algorithm using a large cardinality ensemble under various experimental conditions. These include tracking a reference solution and assimilating Gaussian-distributed noisy observations at various noise levels and observation frequencies. Furthermore, we investigate the performance of the RL agent at assimilating observations with different noise distribution models, namely uniform, log-normal and Gaussian noise. We further explore the RL agent's effectiveness at assimilating partial state observations.

The remaining of the manuscript is organized as follows. Section 2 presents the EnKF algorithm. The RL methodology for DA is then described in Section 3, where an outline of the premise of the framework is first introduced followed by a comprehensive overview of the RL framework presented. Section 4 describes the Lorenz 63 and 96 systems. Section 5 presents the numerical results for tracking and data assimilation using the Lorenz 63 and 96 models. Finally, Section 6 summarizes the main conclusions of this study.

2. Data Assimilation With the Ensemble Kalman Filter

DA is an essential process used in scientific fields such as meteorology, oceanography, and environmental modeling to guide the state of complex systems with incoming observations (Ghil & Malanotte-Rizzoli, 1991; Hoteit et al., 2008). It involves merging observational data with numerical models to enhance predictions once observational information is available (Kalnay, 2002). This process continuously drives the computed system state to align with observations, thereby ensuring accurate and robust state estimates. DA accounts for model and observational uncertainties, offering more reliable predictions for chaotic systems, making it indispensable for tasks such as weather forecasting (Rabier, 2005) and climate modeling (Pedatella et al., 2014). In this study, we adopt the EnKF algorithm, which is the most commonly adopted sequential data assimilation algorithm, as the reference that we benchmark against.

The EnKF algorithm is commonly employed to estimate a discrete-time state process, denoted as $\mathbf{x} = \{\mathbf{x}_n\}_{n\in\mathbb{N}}$, based on observations from a corresponding process $\mathbf{y} = \{\mathbf{y}_n\}_{n\in\mathbb{N}}$. These processes are conventionally connected through a state-space system described as follows:

$$\begin{cases} \mathbf{x}_t = \mathcal{M}(\mathbf{x}_{t-\delta t}) + \mathbf{u}_t \\ \mathbf{y}_t = \mathcal{H}_t(\mathbf{x}_t) + \mathbf{v}_t, \end{cases} \tag{1}$$

where $\mathcal{M}$ represents the nonlinear dynamical model, that advances the system state from time $t - \delta t$ to t, and $\mathcal{H}_t$ the observational operator that projects $\mathbf{x}_t$ from the state space onto the observation space. Here, we make the simplifying assumption that $\mathcal{H}$ is linear, although EnKF algorithms can readily accommodate cases of nonlinear $\mathcal{H}$. The noise terms, $\mathbf{u} = \{\mathbf{u}_t\}_{t\in\mathbb{N}}$ and $\mathbf{v} = \{\mathbf{v}_t\}_{t\in\mathbb{N}}$ are respectively the model and observation process noises. The EnKF algorithm assumes $\mathbf{u}_t$ and $\mathbf{v}_t$ to follow Gaussian distributions with zero means and covariances $\mathbf{Q}_t$ and $\mathbf{R}_t$, respectively. Furthermore, $\mathbf{u}$ and $\mathbf{v}$ are assumed to be independent, jointly independent and independent of the initial state $\mathbf{x}_0$.

The filtering problem involves estimating the state, $\mathbf{x}_t$, based on observations up to time t. EnKF algorithms are primarily designed to provide a MC approximation of the system state distribution using an ensemble of system state realizations. From this ensemble, empirical estimates of the posterior mean state and associated error covariances are derived, typically in the form of sample means and covariances. The process starts with an analysis, denoted by the superscript a, ensemble of size N_{ens} denoted as $\{\mathbf{x}^{a,i}_{t-\delta t}\}_{i=1}^{N_{ens}}$ available at time $t - \delta t$. Subsequently, the forecast, denoted by the superscript f, ensemble at the next time step t is computed by advancing each member $\mathbf{x}^{a,i}_{t-\delta t}$ forward in time using the dynamical model, described as:

$$\mathbf{x}^{f,i}_t = \mathcal{M}(\mathbf{x}^{a,i}_{t-\delta t}) + \eta^i_t, \tag{2}$$

where $\eta^i_t \sim \mathcal{N}(0, \mathbf{Q}_t)$. Upon receiving a new observation $\mathbf{y}_t$, each member of the forecast ensemble is adjusted using the Kalman gain $\mathbf{K}_t$ to generate the analysis ensemble $\left\{\mathbf{x}^{a,(i)}_t\right\}_{i=1}^{N_{ens}}$ according to:

$$\mathbf{x}^{a,i}_t = \mathbf{x}^{f,i}_t + \mathbf{K}_t\left(\mathbf{y}^i_t - \mathcal{H}_t\mathbf{x}^{f,i}_t\right), \tag{3}$$

$$\mathbf{K}_t = \mathbf{P}^f_t\mathcal{H}^T_t\left(\mathcal{H}_t\mathbf{P}^f_t\mathcal{H}^T_t + \mathbf{R}_t\right)^{-1}, \tag{4}$$

where $\mathbf{P}^f_t$ denotes the sample forecast error covariance computed from the forecast members in Equation 2 and $\mathbf{y}^i_n$ represents perturbed observations, that is, $\mathbf{y}^i_t = \mathbf{y}_t + \mu^i_t$ with μ^i_t is a random noise sampled from the observational error distribution. Finally, we note that the RMSE was adopted to evaluate the performance of both DA algorithms, as typically done in previous studies (Anderson, 2001; Bach & Ghil, 2023). This metric also allows for a fair comparison across the considered algorithms since it represents a measure of the error between the prediction and the reference, regardless of the noise model.

3. Data Assimilation With Deep Reinforcement Learning

In RL, agents make sequential decisions to achieve specific goals, with the focus on maximizing cumulative rewards over time (Bertsekas, 2019; Sutton & Barto, 2018). This aligns with decision-making scenarios where actions have consequences, and objectives must be met. RL is particularly relevant to control systems (Azouani & Titi, 2014; Kalantarov & Titi, 2018), where agents learn control policies to influence the behavior of systems (Silver et al., 2014). Hence, adopting an RL framework for DA is a natural progression in the domain, enabling for a nonlinear correction scheme that is also free from restrictive assumptions on the statistics of the observations and model. The key concept in RL is the trade-off between exploration, where the agent experiments with new actions, and exploitation, where the agent chooses known actions with high rewards, mirroring real-world decision-making challenges (Sallab et al., 2017). RL agents learn from feedback, adapt to changing environments, and generalize knowledge to make decisions in new situations.

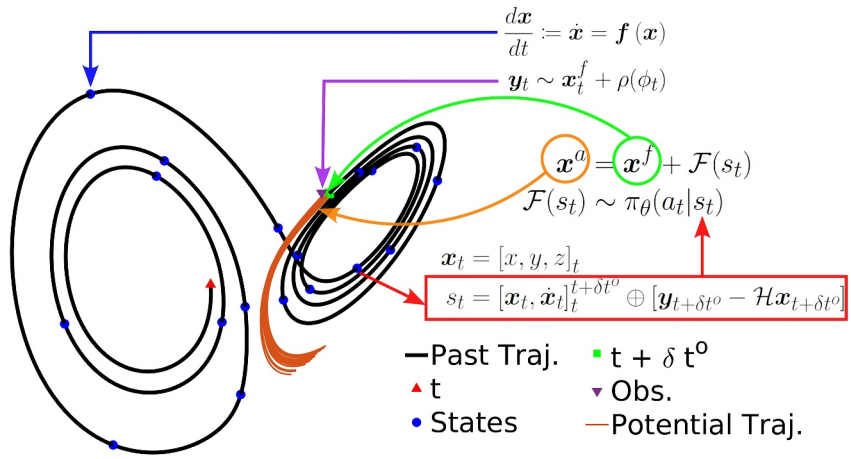


Figure 1. Schematic of the proposed reinforcement learning-based data assimilation framework using the Lorenz 63 as the main example. The plot illustrates the Lorenz 63 solution trajectory (black curve) with an arbitrary assimilation window start time t (red triangle) and corresponding end time $t + \delta t^o$ (green square) when a new observation is available and assimilated. The three dimensional state variables ($\boldsymbol{x}$) of the model are shown at every model time step δt (blue circles). At the last time step, the noisy observational data point ($\boldsymbol{y}$) is shown (inverted purple triangle) alongside the different evolution trajectories (orange curves) following several forecast corrections ($\mathcal{F}(s_t)$) sampled from the policy function $\pi_\theta(a_t|s_t)$. The policy $\pi_\theta(a_t|s_t)$ considers as input state vector the extended state vector composed of the concatenation of the forecast state variables ($\boldsymbol{x}$) and their time derivatives ($\dot{\boldsymbol{x}}$) at each time step δt between t and $t + \delta t^o$ alongside the innovation term, defined as the difference between the observation and its correspondent forecast. Note that the superscripts f and a refer to the forecast and analysis, respectively. The concatenation operation is denoted by $\oplus$, and for the sake of conciseness, concatenation of $\boldsymbol{x}$ and $\dot{\boldsymbol{x}}$ at each δt is represented by the sub- and super-scripts of $[\boldsymbol{x}, \dot{\boldsymbol{x}}]$. Since a stochastic policy is considered in the DA framework, an ensemble of $\mathcal{F}(s_t)$ forecast correction terms are sampled from $\pi_\theta(a_t|s_t)$ when a noisy observation is available. Note that the state variables might not be fully observed, hence the observational operator $\mathcal{H}$ projects the forecast onto the observation space. Moreover, the observation $\boldsymbol{y}$ is considered to be a noisy estimate of the forecast with no restriction on the distribution of the additive noise.

An agent exists in an environment that is described by a set of dynamical rules characterizing its evolution, for example, a system of differential equations (Sutton & Barto, 2018). The agent's responsibility is to make decisions affecting its environment in a way that it maximizes the cumulative reward, or achieves a particular goal. The ultimate outcome of the RL's training procedure is an agent policy $\pi_\theta(a_t|s_t)$, a mapping from the observation space to the action space, which is evaluated to actively control the behavior of the agent at state s_t in a dynamical system. The policy function is generally characterized by a neural network parameterized with θ. Policy functions can be categorized as either deterministic or stochastic; in a deterministic policy, the action with the highest probability is chosen, whereas a stochastic policy relies on random sampling to select an action. In the present framework, a stochastic policy was adopted from which the DA correction term was sampled, where actions are sampled from a Gaussian policy (Schulman et al., 2017). Hence, after training, a policy function is obtained that could be used to sample potential correction terms from a distribution that adapts to the agent's state, and allowing to generate an ensemble of states via MC sampling. In contrast with most efforts put for developing efficient DA schemes; for example, (Buizza et al., 2022; Farchi et al., 2021; Lermusiaux, 2007), the RL machinery relies on a nonlinear neural network to provide a correction without being restricted to a pre-computed data set for supervising its training. Furthermore, the RL agent does not require any assumption on the noise distribution of the observational errors nor restrictive assumptions on the model.

In this study, the agent receives information, in the form of an extended state vector describing the system, denoted by states (s_t), that includes the forecast states and their derivatives $\boldsymbol{x}^f$ and $\dot{\boldsymbol{x}}^f$, respectively, at each model time step δt starting from the time t of the previous observation till the next observational time step $t + \delta t^o$, and the innovation term $\boldsymbol{y} - \mathcal{H}\boldsymbol{x}^f$. A schematic diagram of the framework is illustrated in Figure 1 and the corresponding formulation is further expanded upon in Section 3.3. Note that the notation followed in the RL methodology is similar to that of the EnKF, where the $\boldsymbol{x}^f$ and $\boldsymbol{y}$ are those from Equation 1. This extended state formulation could also be modified to tackle the high dimensional setting in which operating on an augmented state could be

expensive. Significant reduction could be achieved by employing the Markovian assumption, such that only the last forecast state, its derivative and the innovation term could be used as input to the policy function.

The agent interacts with the environment to change its course of evolution and adapts to these changes to maximize the cumulative reward, as later defined, gathered over some period of time (Silver et al., 2014). This interaction is formulated mathematically as:

$$\boldsymbol{x}^a = \boldsymbol{x}^f + \mathcal{F}(s_t), \tag{5}$$

where the corrected state vector, $\boldsymbol{x}^a$ is the sum of the model forecast, $\boldsymbol{x}^f$, and the forecast correction term $\mathcal{F}(s_t)$, which is sampled from $\pi_\theta(a_t|s_t)$. This $\mathcal{F}$ is the nonlinear forecast correction term that updates the forecast state using incoming observational data. Note that the linear update equation is kept similar to that of the EnKF algorithm, which relies on a linear update term (Hoteit et al., 2008). In the current configuration, the RL agent is not provided with statistical information regarding the noisy observations. Instead, it employs an MC strategy, using an RL agent that employs random stochastic policy sampling. This approach generates an ensemble of assimilated solutions, which are subsequently averaged to produce an improved estimate of the system state, denoted by RL-50 in the following sections.

The training cycle is defined by specifying the reward function (Lillicrap et al., 2015). We test out several reward functions in this preliminary investigation, where the agent's performance was evaluated using the mutual information, negative of the RMSE and RMSE^{-1}. While these reward functions are mathematically similar (Guo et al., 2005; Seidler, 1971), the associated training stability is different. Accordingly, the reward terms were selected as the negative of the RMSE, which strikes a satisfactory balance between interpretability, computational cost and agent's performance. Finally, we note that different reward functions result in different optimality criteria, allowing for task-specific RL model calibration.

3.1. Reinforcement Learning

The RL framework involves training an agent through several interactions with the environment, in the present context, the dynamical system. Training an RL agent requires a large number of interactions with the environment and consequently a large unavoidable computational load often several orders of magnitude greater than solving the underlying differential equations. Despite the computational expenses associated with RL, its distinctive advantages, unparalleled by alternative approaches, render such costs justifiable. RL is capable of handling very complex decision-making environments with high-dimensional state and action spaces that are infeasible for traditional methods. Unlike traditional deep learning techniques, RL agents enhance their generalization skills through direct interactions with the environment (Novati et al., 2021). A notable strength of RL is its adaptability, where the RL agent can adjust to evolving environments and learn new tasks without being explicitly programmed (Mnih et al., 2013).

On the bright side, the field of RL has become more accessible in recent times, thanks to open-source libraries like `smarties` (Novati & Koumoutsakos, 2019) and `stable baselines3` (Raffin et al., 2021), among others. Furthermore, recent algorithms, such as Proximal Policy Optimization (PPO), are less sensitive to the algorithm's associated hyperparameters in comparison to earlier RL algorithms. In this work, we leverage the capabilities of `stable baselines3`, a high-performance RL software designed to exploit parallel computing, distributing the training process across multiple computational nodes. Each node simulates a distinct trajectory of the Lorenz 63 system, providing a large set of agent-environment interactions that are used to train the agent. In this parallelized setup, each computational node accumulates experiences by independently interacting with various instances of the environment. These experiences are then structured into episodes defined as:

$$\tau = \{s_t, r_t, a_t, s_{t+1}\}_{0:T}, \tag{6}$$

where τ is the ordered set of interactions across a time horizon, t represents the time at which the environment is at state s_t, a_t is the action the agent takes at that time, r_t is the reward the agent receives from performing action a_t and s_{t+1} is the subsequent state.

The RL agent's training objective is to maximize the expected cumulative discounted reward function, defined as:

$$R_t = \sum_{t=0}^{T} \gamma^t r_t, \tag{7}$$

where $\gamma \in [0, 1)$ is the discount factor. In our specific setting, a smaller value of γ proves advantageous, given the random noise sampling. This choice of reducing the emphasis on distant future rewards results in a more stable agent performance.

The policy function π_θ is a mapping between the agent's state and the action space, which can be structured either as a set of discrete actions or as a probability distribution function for continuous actions. As previously mentioned, policy functions are either deterministic, the action to most likely result in the highest reward is chosen, or stochastic, where actions are randomly sampled from a distribution that is typically approximated by a surrogate model. Here, the policy π_θ is represented as a densely connected multi-layer perceptron (Chen & Chen, 1995) parameterized by θ. Furthermore, actions assume continuous values, leading π_θ to output a probability distribution over possible actions.

3.2. Proximal Policy Optimization

The PPO algorithm (Schulman et al., 2017) trains an agent using two key components, each parameterized by distinct neural networks: an actor network that takes the environment's state as input and produces the corresponding action, and a critic network that also takes the environment's state as input and predicts the discounted reward (Mnih et al., 2016). In our study, both the actor and critic networks are represented by multi-layer perceptrons, each composed of two hidden layers, each containing 128 neurons. For spatially varying systems, the actor and critic networks could be modeled using convolutional networks to better extract the spatial dependencies in the state variables.

The essence of the PPO algorithm revolves around optimizing the actor network to maximize the cumulative reward obtained by the agent, and the critic network to minimize the mean squared error between the predicted and actual expected cumulative rewards, starting from a given state. This optimization can be mathematically expressed through two distinct loss functions that are fully described in the following. The actor network is optimized by maximizing the actor's objective function:

$$J_{actor} = \mathbb{E}\left[\min\left(q_t(\theta)\hat{A}_t, \text{clip}\left(q_t(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_t\right)\right], \tag{8}$$

where $q_t(\theta) = \pi_\theta(a_t|s_t)/\pi_{old}(a_t|s_t)$ is the ratio of the probability of adopting an action a_t at state s_t using π_θ to that of the previous policy π_{old}, and $\hat{A}$ is the advantage (Mnih et al., 2016), which quantifies how favorable the observed outcome of selecting a particular action is compared to the estimated discounted reward of the current state. The advantage is described as:

$$\hat{A} = V_{target} - V_{\theta,old}, \tag{9}$$

where, $V_{target} = \sum_{i=0}^{T-1} r_i\gamma^i + \gamma^T V_{\theta,old}(s_T)$ is the discounted reward computed using the agent's interactions with the environment and $V_{\theta,old}$ is the discounted reward predicted by the critic network.

Furthermore, the present setting relies on policy clipping with a clipping coefficient $\epsilon = 0.2$ (Schulman et al., 2017), where $q_t(\theta) \in [1-\epsilon, 1+\epsilon]$. This policy clipping mechanism helps maintain policy stability during parameter updates, stabilizing the training process. On the other hand, the critic loss is given as:

$$L_{critic} = \mathbb{E}\left[\hat{A}^2\right], \tag{10}$$

where, $\mathbb{E}$ is the expectation operator. Finally, the total loss function used to train the RL agent is given as:

$$L = \mathbb{E}\left[\min\left(q_t(\theta)\hat{A}_t, \text{clip}\left(q_t(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}_t\right)\right] - v_f\mathbb{E}\left[\hat{A}^2\right], \tag{11}$$

where v_f is a positive constant called the value function coefficient.

3.3. Data Assimilation Using Reinforcement Learning

We explore a novel data assimilation framework that leverages RL to assimilate noisy observations for improving model forecasts. The RL agent receives noisy information about the system's state variables, and its policy, $\pi_\theta(a_t|s_t)$ that is contingent upon the environment's state s_t takes an action according to the preassigned strategy. The state upon which the agent's policy is evaluated consists of the extended vector composed by the concatenation $\left[\boldsymbol{x}^f, \dot{\boldsymbol{x}}\right]_t^f \oplus \left[\boldsymbol{x}^f, \dot{\boldsymbol{x}}\right]_{t+\delta t}^f \oplus \ldots \left[\boldsymbol{x}^f, \dot{\boldsymbol{x}}\right]_{t+\delta t^o}^f \oplus \left[\boldsymbol{y}_{t+\delta t^o} - \mathcal{H}\boldsymbol{x}_{t+\delta t^o}^f\right]$. Notably, this selection preserves the Markovian assumption inherent in the EnKF, as $\mathcal{F}\left(\boldsymbol{x}|_t^{t+\delta t^o}\right) = \mathcal{F}(\boldsymbol{x}(t+\delta t^o))$. However, including forecast information from previous steps significantly enhances training stability, even though it comes at the cost of a higher dimensional input. This formulation might not be the most efficient for high dimensional problems, but it could be alleviated by assuming that the system is Markovian. This gives rise to the question of how long back-in-time should forecast states be considered. In particular, the Markovian assumption enables the use of the last state as input to the RL agent, as opposed to using all the forecast steps. This leads to a trade-off between memory efficiency and stability in training, where additional information from the forecast improve the performance of the RL agent (Mnih et al., 2013).

In the present context, we introduce an RL agent responsible for correcting model forecasts of the dynamical system states using the update equation:

$$\boldsymbol{x}_{t+\delta t^o}^a = \boldsymbol{x}_{t+\delta t^o}^f + \mathcal{F}\left(\boldsymbol{x}|_t^{t+\delta t^o}, \dot{\boldsymbol{x}}|_t^{t+\delta t^o}, \boldsymbol{y}_{t+\delta t^o} - \mathcal{H}\boldsymbol{x}_{t+\delta t^o}^f\right), \tag{12}$$

where $\mathcal{F}$ represents the RL agent's policy π_θ, and is written in the fully expanded form here, whereas later a shorthand notation is adopted. The policy takes as input the state vector $\boldsymbol{x}$ and the first-order derivatives $\dot{\boldsymbol{x}}$ at all time steps from t to $t+\delta t^o$ at δt increments, as well as the innovation term $\boldsymbol{y} - \mathcal{H}\boldsymbol{x}^f$. Since a stochastic policy function is considered, the study examines the performance of a single RL agent by taking maximum probability action, and the performance of an ensemble of agents by randomly sampling the policy function for actions.

3.4. Training the DA Agent

The experimental setup encompasses various hyper-parameters that require tuning to achieve satisfactory performance. The parameters subjected to tuning include the learning rate, γ, number of assimilation steps per episode ($n_{a,train}$), total number of episodes, v_f, ϵ. Experiences have shown that the performance of a stable agent is most sensitive to γ, v_f and gradient clipping.

The process of hyper-parameter optimization commenced with a Latin hypercube sampling strategy to establish a baseline assessment of the acceptable range of values for these parameters. Subsequently, the training process is repeated using a new set of hyperparameters selected from within a finer-scale parameter space. For all experiments conducted, we employed the ADAM stochastic optimization algorithm (Kingma & Ba, 2017) to optimize the loss function for the parameters of the actor and critic networks. The parameters utilized for training the agents, which underpin the results presented in this study, are detailed in the Supplementary.

The RL agent's training objective centered on maximizing the cumulative rewards accrued over a specific time horizon. At each assimilation step, the reward was calculated as the negative RMSE between the observation and the forecast generated by the preceding action. This choice was made because minimizing the RMSE is equivalent to maximizing the mutual information between the compared quantities and because the RMSE is ultimately the measure that is used to evaluate the performance of the agent (Guo et al., 2005; Rodriguez, 2021). More specifically, because the experiments in this study feature a well-defined reference solution obtained through twin experiments, we report the RMSE of both the RL and EnKF solutions with respect to the noise-free reference solution that was obtained through a twin experiment. The RMSE hence provides quantitative estimates that help examine the assimilated solution in terms of forecast and analysis. Finally, we note that the RL agents are trained for the same experimental conditions (i.e., partial observations, noise level, noise model and observation frequency) as the ones used for evaluation, but, the initial conditions used for evaluation are far from those used for training.

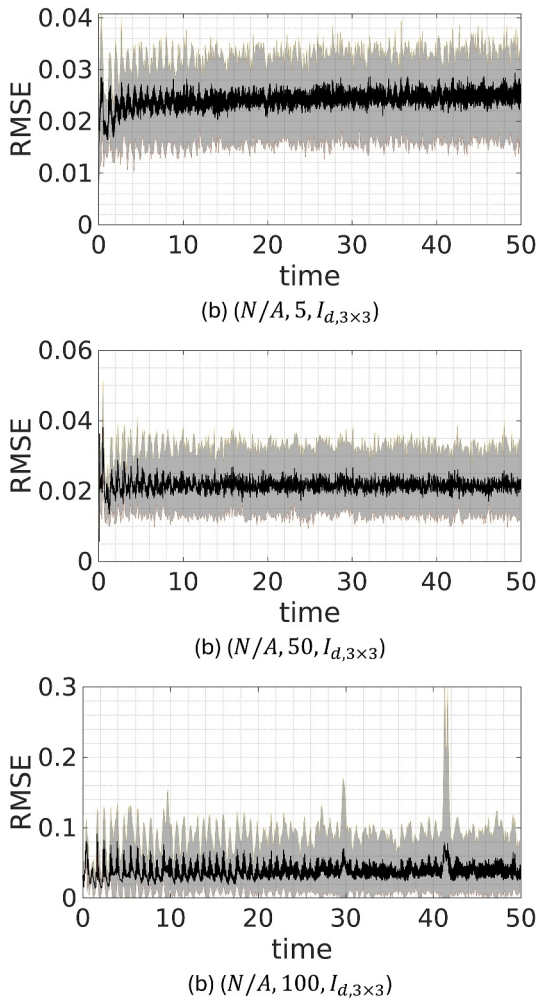


Figure 2. Evolution of the mean RMSE (solid lines) and its $\pm\sigma$ (shadowed) based on 50 experiment repetitions with the Lorenz 63 model. Plotted are results for tracking a noise-free reference solution. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

4. Experimental Setup

4.1. Lorenz 63

The Lorenz 63 is a set of three deterministic ordinary nonlinear differential equations developed to simulate simplified atmospheric convection (Lorenz, 1963). This system is renowned for its manifestation of chaotic behavior, where even minuscule perturbations in initial conditions lead to substantially divergent solution trajectories over time (Eckmann & Ruelle, 1985; Hammoud et al., 2022). The Lorenz equations have been extensively studied in chaos theory and nonlinear dynamics, and have been the fundamental benchmark to develop new data assimilation techniques (Foias et al., 2001; Hayden et al., 2011). The Lorenz 63 equations are given by:

$$\dot{x} = \sigma(y - x), \tag{13}$$

$$\dot{y} = x(\rho - z) - y, \tag{14}$$

$$\dot{z} = xy - \beta z, \tag{15}$$

where, σ, ρ, and β are typically positive constants. This system is known to exhibit a chaotic attractor for $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$. In this study, the system of equations were solved using an 2nd order Runge-Kutta scheme with a time step $\delta t = 0.001$, which offers a suitable balance between solution accuracy and computational time for the application at hand.

4.2. Lorenz 96

The Lorenz 96 is a simplified yet insightful model for understanding the nonlinear processes inherent in atmospheric circulation. The dynamical model is characterized by a system of ordinary differential equations, it encapsulates the temporal evolution of some atmospheric state variables distributed along a hypothetical latitude circle. Each of the Lorenz 96 variables, denoted by X_k, is subject to the influences of its immediate neighbors in a cyclically continuous manner. This system has been extensively adopted to benchmark data assimilation algorithms; for example, (Howard et al., 2024; Raboudi et al., 2023). The dynamical model is represented by the following governing equations:

$$\frac{dX_k}{dt} = -X_{k-1}(X_{k-2} - X_{k+1}) - X_k + F, \tag{16}$$

where F serves as an external forcing term. Despite its abstract simplicity, the Lorenz 96 model yields complex, chaotic behaviors that mimic those of real-world atmospheric dynamics when the forcing term is sufficiently large. Typically, the system transitions to chaotic for $F = 8$, which has become a classical setting for studying chaotic dynamics with the Lorenz 96. The numerical results are shown for the 40-dimensional Lorenz 96 with periodic boundary conditions and $F = 8$. The system of equations were solved with a 4th order Runge-Kutta scheme with a time step $\delta t = 0.05$.

Two primary strategies are commonly adopted in the RL literature when dealing with high-dimensional problems. First, the Markovian assumption could be applied, where instead of augmenting the observed states with prior forecast states, the last few could be used as input (Mnih et al., 2013). With the Lorenz 96 case, only the last forecast states, its derivative and the innovation term are used as input, which reduces the dimensionality of the input by $\mathcal{O}(\delta t^o/\delta t) = \mathcal{T}$, where $\mathcal{T}$ is called the assimilation window size. A second approach, which deals with spatially varying fields, would be to employ a more efficient deep learning model to represent the policy function. For instance, convolutional, recurrent and graph neural networks would allow for significant dimensionality reduction, and are commonly adopted for RL tasks (e.g., Jumper et al., 2021; Mnih et al., 2015).

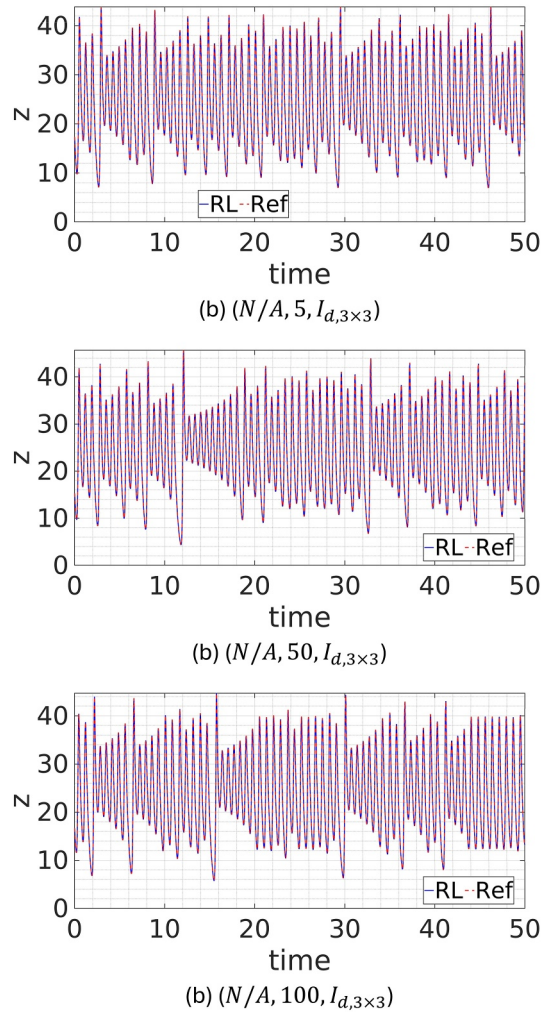


Figure 3. Evolution of the z-variable for a sample RL solution (solid blue lines) and corresponding reference (dashed red line). Plotted are results for tracking a noise-free reference solution. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

5. Results

The RL-DA framework is systematically assessed under different experimental conditions. First, an RL agent was trained to track a reference solution of the Lorenz 63 system using coarse-in-time, noise-free observations of all state variables. Here, the objective is to investigate whether the RL solution can be synchronized with a noise-free reference solution when a stochastic policy function is employed. In the second setting, a more practical setting was adopted, where an ensemble of noisy observations are assimilated to improve the model forecast. In particular, we investigate the performance of the RL and EnKF algorithms at data assimilation under various experimental conditions using the Lorenz 63 and 96 models.

5.1. Tracking Reference Solutions

Here, we investigate whether the RL agent can maintain a reasonably close solution in comparison to the reference, and prevent them from diverging. Focus is set on the Lorenz 63 as a simplified example, where additional attention is set on the Lorenz 96 system in the case of data assimilation with noisy observations. Three training regimes were explored, involving observations every 5, 50, and 100 δt, corresponding to δt^o of 0.005, 0.05, and 0.1 time units, respectively. Evolution curves of the RMSEs of the RL solutions are presented in Figure 2. The average RMSE is represented by a solid black line, encircled by a shaded region denoting one standard deviation ($\pm\sigma$), based on 50 repetitions of the experiment involving different reference solutions. The

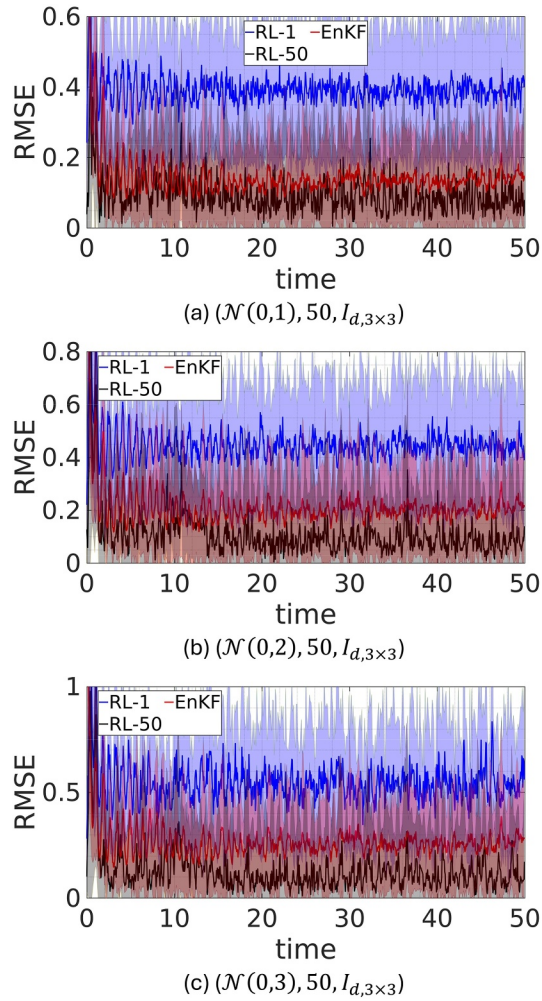


Figure 4. Evolution of the mean RMSE (solid lines) and its $\pm\sigma$ (shadowed) based on 50 experiment repetitions with the Lorenz 63 model. Plotted are results for assimilating noisy observations for increasing noise levels. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

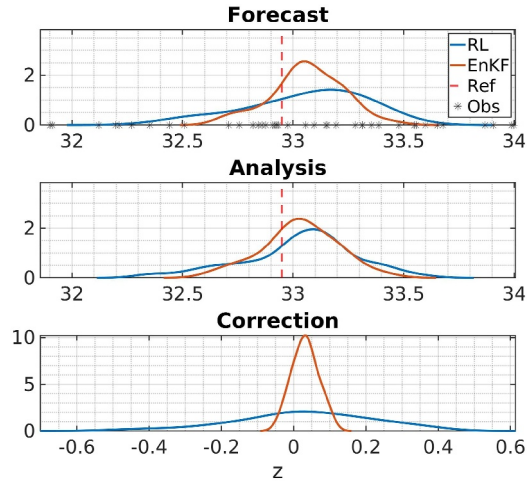
plots indicate that the RMSE is on average approximately 0.025 for assimilation window sizes $\mathcal{T}$ of 5 and 50, and increase to 0.05 for $\mathcal{T} = 100$. Figure 3 illustrates RL and reference solutions for the z-variable in the Lorenz 63 system, based on randomly selected reference trajectories. These curves highlight strong agreement between the RL solution and the reference, further corroborating the results presented in Figure 2.

5.2. Assimilating Noisy Observations

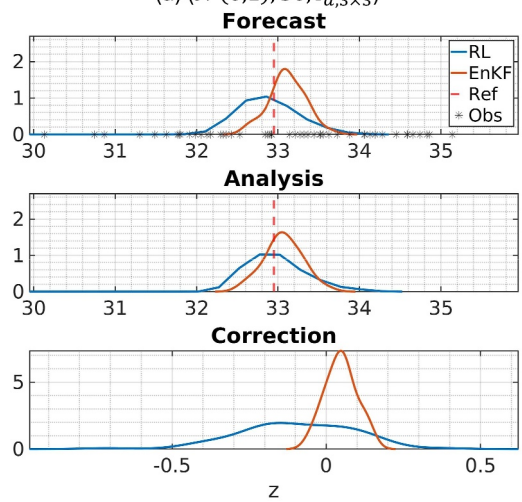
In a realistic scenario, an ensemble of noisy observations are assimilated to improve the model forecast. We explore the influence of noise levels (σ), $\mathcal{T}$, statistical noise distribution, and partial state observability on the RL agent's performance in the cases of the Lorenz 63 and 96 dynamical models. Moreover, we conduct a comparative analysis by benchmarking the outcomes of the RL approach with those of the EnKF, relying on an ensemble comprising 50 members. To ensure robustness and build confidence in the results, each of the RL and EnKF experiments was repeated 50 times using different reference solutions, providing reliable estimates of the RMSE.

5.2.1. Noise Level

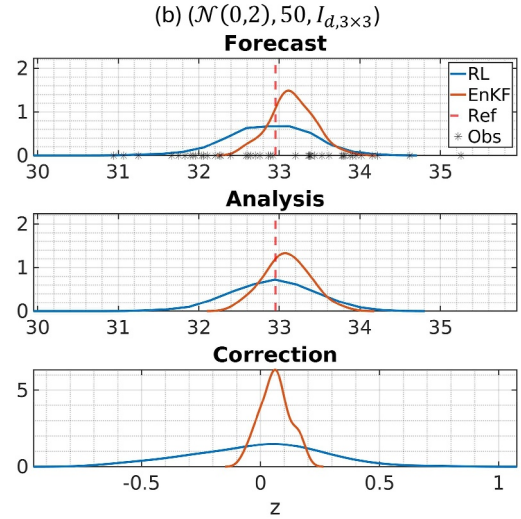
We first consider the Lorenz 63 system and examine the scenario of fully observed state available at regular intervals of $\mathcal{T} = 50$, with additive noise drawn from a Gaussian distribution characterized by zero mean and



(a) $(\mathcal{N}(0,1), 50, I_{d,3\times3})$



(b) $(\mathcal{N}(0,2), 50, I_{d,3\times3})$



(c) $(\mathcal{N}(0,3), 50, I_{d,3\times3})$

Figure 5. Evolution of the z-variable for a sample RL solution (solid blue lines) and corresponding reference (dashed red line). Plotted are results for tracking a noise-free reference solution. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the observation window size and $\mathcal{H}$ the observational operator.

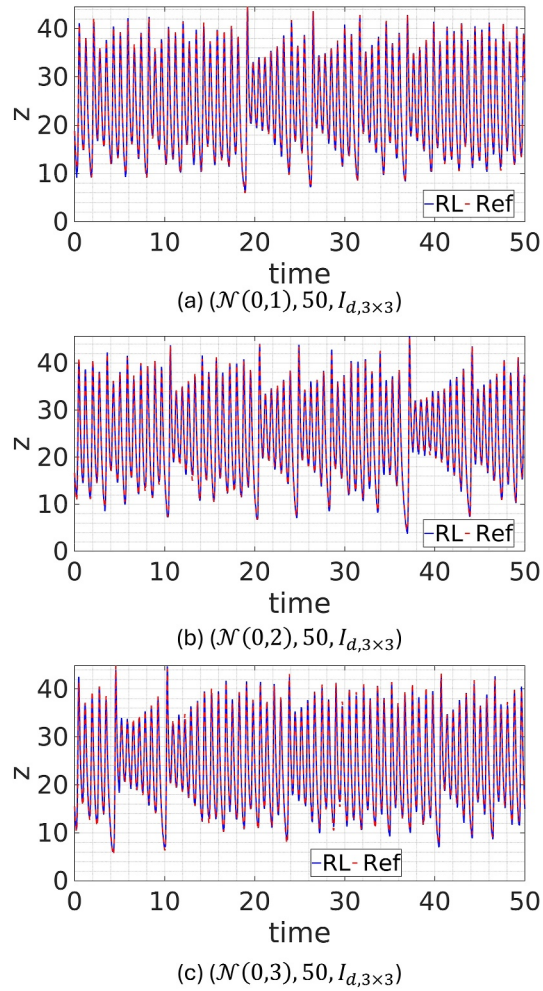


Figure 6. Evolution of the z-variable for a sample RL solution (solid blue lines) and corresponding reference (dashed red line) for the Lorenz 63 system. Plotted are results for tracking a noise-free reference solution. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the observation window size and $\mathcal{H}$ the observational operator.

standard deviation σ. We investigate the influence of varying σ on the assimilated solution by computing the RMSE at the analysis step. We compare the results obtained from a single RL agent, an average solution derived from 50 distinct RL trajectories with actions randomly sampled from the agent's policy, and the EnKF solution based on an ensemble comprising 50 members. Note that this comparison places the RL agent at a disadvantage, as it was trained without any statistical information about the response of the system to observation noise. Nonetheless, we believe that the comparison with the EnKF prediction is meaningful as it represents the primary benchmark against which DA algorithms are evaluated, despite the more suitable comparison with the Kalman Filter. This also highlights potential limitations that will be addressed in future studies, where information about the statistics of the ensemble would be exploited to implement a more sophisticated RL strategy for DA. Notably, our algorithm consistently outperforms the Kalman Filter across all experiments and hence not shown.

Figure 4 presents the RMSE evolution over time for the assimilated solution, resulting from RL and EnKF under different σ values for the case of the Lorenz 63. The plots suggest that, across all σ values considered, the EnKF solution exhibits slightly lower RMSE values than those of a single RL agent, and slightly larger RMSEs than the RL solution obtained by averaging 50 action realizations. This observation yields two significant insights: first, the potential computational efficiency gain from employing a single RL agent for DA. Second, using a single RL agent with a stochastic policy allows for sampling a diverse set of forecast corrections, providing a new ensemble of state estimates that when averaged, generally results in a lower RMSE compared to an EnKF solution produced

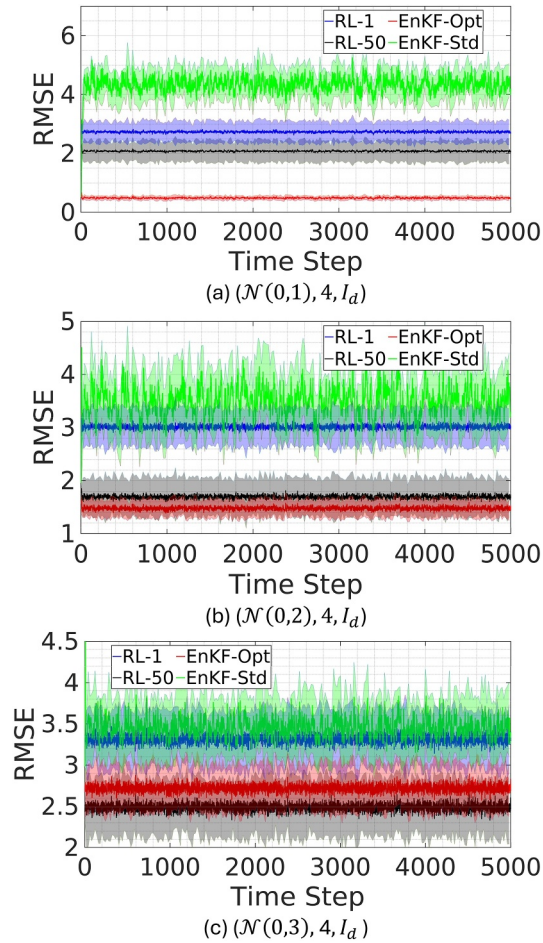


Figure 7. Evolution of the mean RMSE (solid lines) and its $\pm\sigma$ (shadowed) based on 50 experiment repetitions with the Lorenz 96 model. Plotted are the EnKF and RL results for assimilating noisy observations for increasing noise levels. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

using an equivalent ensemble size. A definitive statement on the computational advantages of RL is premature, requiring a computational study to develop a scalable RL code and benchmark it against a scalable DA code.

Figure 5 illustrates the transition of the PDF after the correction is made alongside the distribution of the corrections for the RL and EnKF for the Lorenz 63 case. The results indicate that the RL distribution is wider and covers more of the observations than the EnKF, meaning that the RL ensemble is richer in terms of information it provides even though individual realizations perform poorer than the EnKF solution. This suggests that the RL ensemble is able to capture more diverse events that are far from the mean solution, and is hypothesized to act similar to covariance inflation (Luo & Hoteit, 2011). On the other hand, the mean of the RL solutions is closer to the reference solution than the average EnKF solution, aligning well with the results obtained earlier. The plot also shows the distribution of the corrections, indicating that the distribution for the RL corrections is wider than that of the EnKF and suggesting that the EnKF is comparatively conservative when performing updates. Similar results for the remaining experiments are analyzed in Supporting Information S1.

As σ increases, noticeable high-amplitude, abrupt variations in RMSE are observed in the assimilated solutions, and the time-averaged RMSE increases. In the second row of Figure 6, we present the RL and reference evolution curves corresponding to the z-variable. The results suggest that the RL solution closely follows the reference solution for all σ values considered. However, as σ increases, slight deviations between the RL solution and the reference become evident, particularly at the peaks and troughs of the curves. Nevertheless, the RL agent successfully assimilates noisy data, at high noise levels.

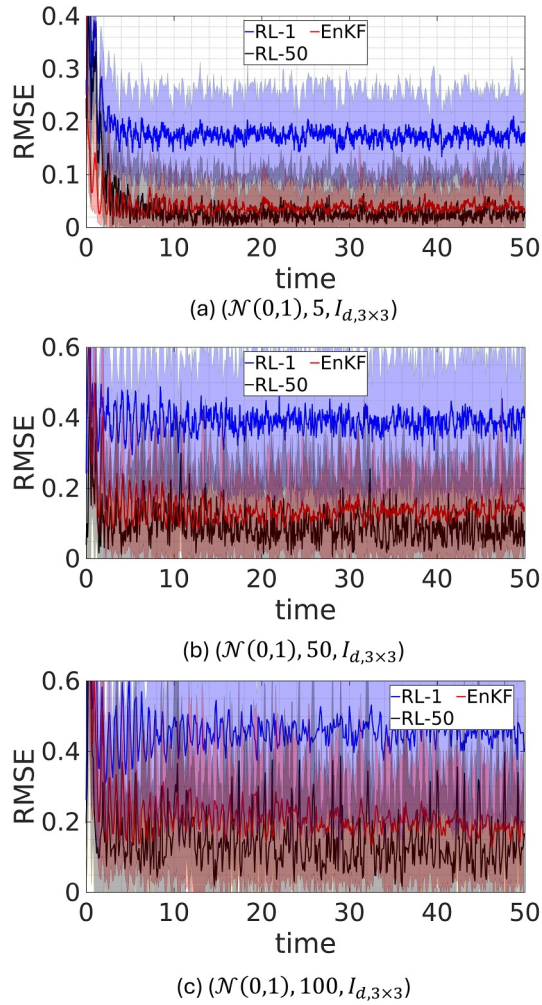


Figure 8. Evolution of the mean RMSE (solid lines) and its $\pm\sigma$ (shadowed) based on 50 experiment repetitions with the Lorenz 63 model. Plotted are results for assimilating noisy observations for increasing $\mathcal{T}$. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

Figure 7 illustrates the evolution of the RMSE as a function of the assimilation time step for the Lorenz 96 case. The RRMSE curves are shown for RL-1, RL-50, standard EnKF and the EnKF with localization and inflation. The plot indicates that for Gaussian noise with $\sigma = 1$ and $\mathcal{T} = 4$ (1 day), the RMSE of the standard EnKF is largest, however, when the EnKF is equipped with inflation and localization, it achieves the lowest RMSE among all experiments. On the other hand, both RL-1 and RL-50 achieve a lower RMSE in comparison to the standard EnKF, however, cannot surpass the performance of the optimized EnKF. This underscores the importance of developing the DA-RL framework to exploit information about the statistics of the ensemble, and employ techniques such as covariance inflation and localization. On the other hand, as the noise level increases, the performance of the RL-50 get closer to that of the optimized EnKF, surpassing it for $\sigma = 3$. The results also show that RL-1 and RL-50 perform better than the standard EnKF with the RL-50 benefiting from the additional information obtained by random sampling of the policy function.

5.2.2. Assimilation Window Size

Observational data may often become available at varying assimilation window sizes, necessitating a DA scheme capable of accommodating different observation rates. In light of this requirement, we trained an RL agent to assimilate noisy data for distinct $\mathcal{T}$, thereby examining the influence of high-frequency ($\mathcal{T} = 5$), medium-frequency ($\mathcal{T} = 50$), and low-frequency ($\mathcal{T} = 100$) observations. The middle row of Figure 8 depicts the

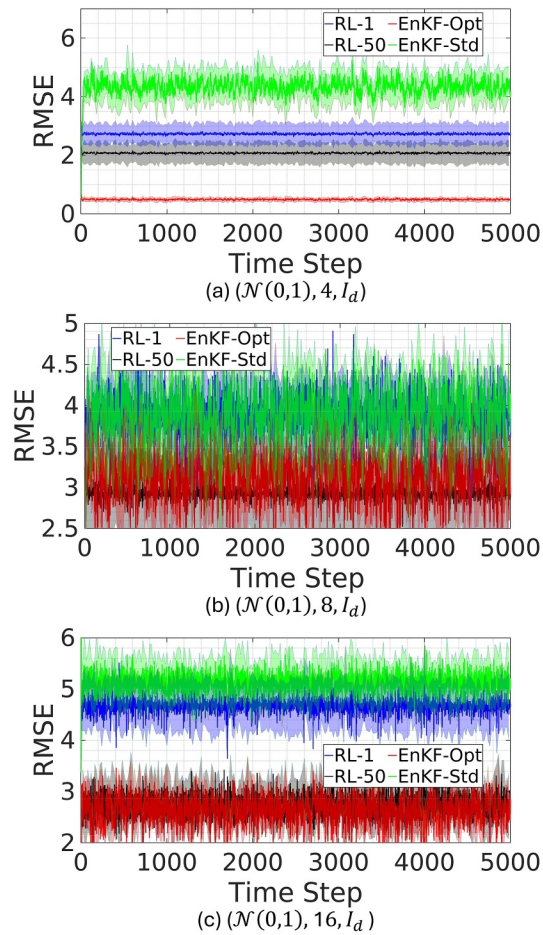


Figure 9. Evolution of the mean RMSE (solid lines) and its $\pm\sigma$ (shadowed) based on 50 experiment repetitions with the Lorenz 96 model. Plotted are the EnKF and RL results for assimilating noisy observations for increasing $\mathcal{T}$. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

progression of RMSE under varying $\mathcal{T}$ for the Lorenz 63 case. Across all considered $\mathcal{T}$, the results suggest that a single RL agent exhibits slightly larger RMSE compared to those achieved by the 50-member EnKF solution. For all cases, the 50 RL agent-averaged solution demonstrates a lower time-averaged RMSE in contrast to the 50-member averaged EnKF solution. This indicates that even when the RL agents do not communicate among each other, an MC averaged solution achieves lower RMSEs than the EnKF solution with 50 members. Nevertheless, these results underscore the need to develop more sophisticated RL approaches, potentially utilizing multi-agent RL (Albrecht et al., 2023), that incorporate ensemble information when performing the correction step.

Figure 9 presents the evolution curves of the RMSE against the assimilation time step for the case of the Lorenz 96. The plots show the results for RL-1 and RL-50, and compares them to those of the standard EnKF and the EnKF employing localization and inflation. Here, the RL agent is trained to assimilate noisy data for $\mathcal{T} = 4$, 8 and 16; equivalent to 1, 2, and 4 physical days with normally distributed noisy observations with unit variance. The plots indicate that for $\mathcal{T} = 4$, the RMSE corresponding to the EnKF equipped with localization and inflation is the lowest, whereas that of the standard EnKF is the largest. The plot also indicates that the RL-50 achieves a lower RMSE in comparison to the single agent RL. As $\mathcal{T}$ increases, however, the performance of RL-50 approaches that of the optimized EnKF, such that both perform similarly, whereas the RL-1 and standard EnKF perform worse than the optimized EnKF and the RL-50 methods. Nevertheless, it is worthy to note that RL-50 is based on randomly sampling the policy function, which does not take into account the ensemble statistics nor does it employ inflation and localization techniques. This particular aspect will be at the core of our future work.

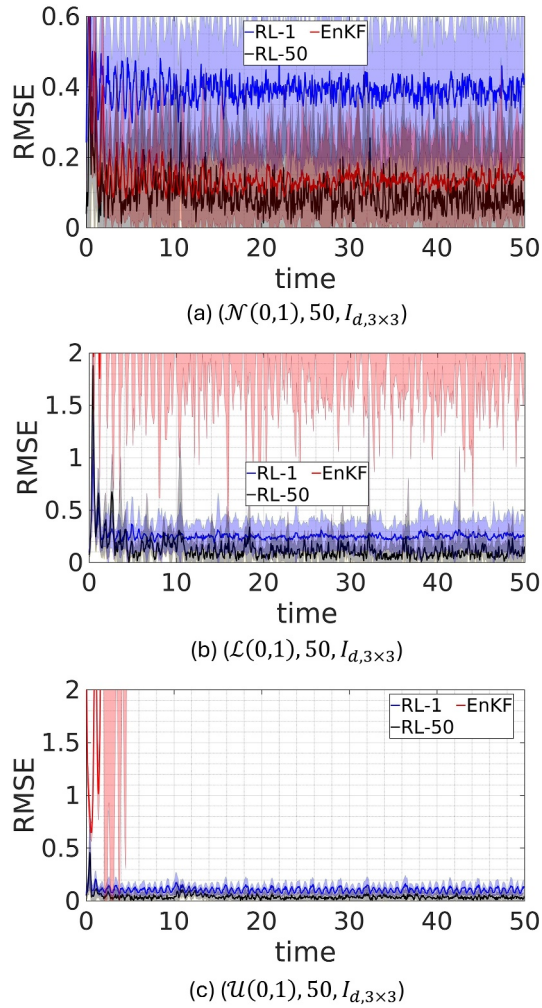


Figure 10. Evolution of the mean RMSE (solid lines) and its $\pm\sigma$ (shadowed) based on 50 experiment repetitions with the Lorenz 63 model. Plotted are the EnKF and RL results for assimilating noisy observations for different noise models. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

5.2.3. Noise Distribution

A major limitation of the EnKF is its reliance on normally distributed observations of system states. We investigate the impact of different statistical distributions of observations on the DA performance of the RL agent. Specifically, we examine cases involving unbiased standard Gaussian ($\mathcal{N}$), strongly positively biased standard log-normal ($\mathcal{L}$), and weakly positively biased standard uniform observational noise ($\mathcal{U}$). Figure 10 presents the evolution curves of the RMSE for various observational noise distributions for the Lorenz 63 case. The plots illustrate that for the case of standard Gaussian noise, both the single RL agent and EnKF solutions effectively assimilate noisy observational data with a slightly lower RMSE value achieved by the EnKF solution. On the other hand, the 50-realization averaged RL solution yields a lower RMSE compared to the 50-member EnKF solution. For log-normal and uniform noise distributions, the EnKF experiences large errors when assimilating noisy observations. Conversely, a single RL agent successfully assimilates these noisy observations, providing an assimilated solution that is close to the reference solution. Further improvements are observed when averaging the solutions obtained through policy sampling across 50 different realizations. Figure 11 presents the RL and reference evolution curves for the z-variable. The plots indicate that the RL solution follows the reference solution reasonably well for all the noise distributions that were considered. The curves clearly illustrate that the RL agent is able to assimilate non-Gaussian noisy observations even when observations are perturbed with biased noise.

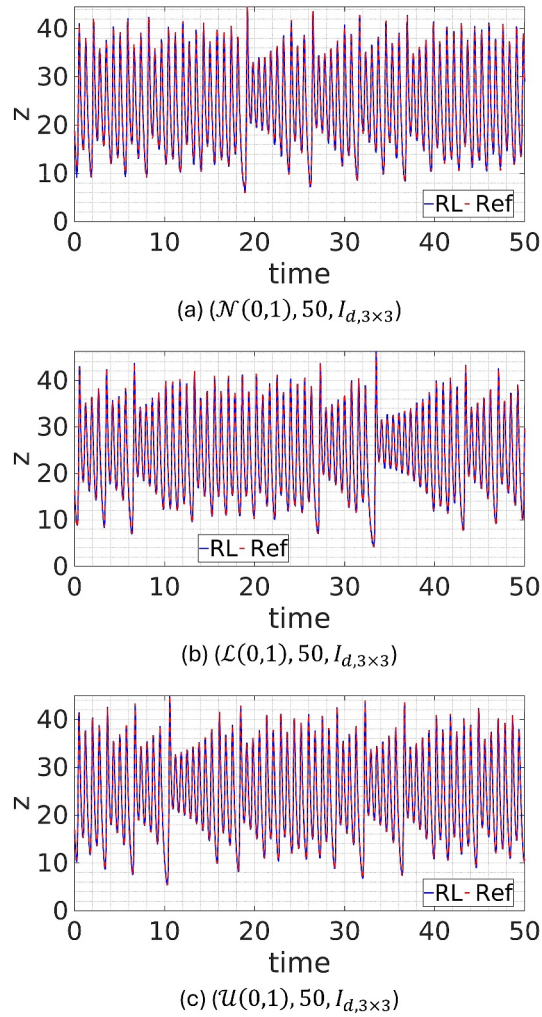


Figure 11. Evolution of the z-variable for a sample RL solution (solid blue lines) and corresponding reference (dashed red line) for the Lorenz 63 system. Plotted are the EnKF and RL results for assimilating noisy observations for different noise models. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

Figure 12 illustrates the evolution curves of the RMSE in terms of the assimilation time step as resulting from RL-1, RL-50, the standard EnKF and the EnKF with localization and inflation. Here, the RL agent is trained to assimilate noisy data from standard normal, lognormal and uniform distributions for $\mathcal{T} = 4$. The plots indicate that for a Gaussian noise model, the lowest RMSE is achieved by the optimized EnKF equipped with localization and inflation, whereas the largest RMSE corresponds to the standard EnKF. The RL-50 achieves a lower RMSE in comparison to the single agent RL, but remains slightly larger than that of the optimized EnKF. For the lognormal and uniform noise models, the EnKF (not plotted) faces numerical instability leading to numerical blowup. On the other hand, the RL algorithm remains stable and achieves RMSE values that are comparable with those obtained in the case of normally distributed noise. Specifically, the RL-50 errors normalized by the norm of the reference solution are approximately 30% for the Gaussian and lognormal noise and 15% for the case of uniformly distributed observations. Even though this error might seem high, we reiterate that we are presenting a proof-of-concept showcasing the potential of RL in application to DA. We further emphasize that the current framework serves as a corner stone for a new approach in nonlinear DA, where additional developments could be built on top of this initial study. In particular, exploiting information about the statistics of the ensemble members, employing localization and inflation techniques, and applying more sophisticated deep learning strategies to model the policy function are all open research avenues that will be explored in future work.

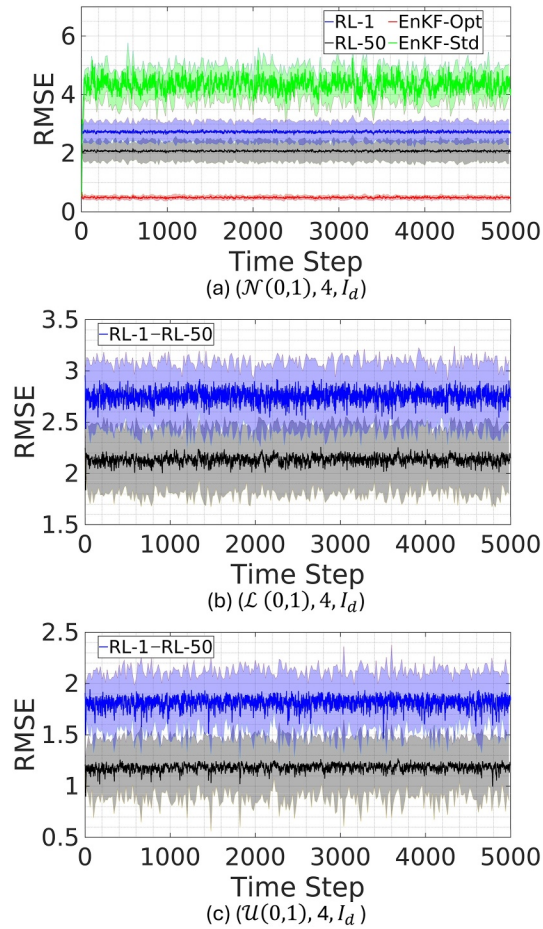


Figure 12. Evolution of the mean RMSE (solid lines) and its $\pm\sigma$ (shadowed) based on 50 experiment repetitions with the Lorenz 96 model. Plotted are the EnKF and RL results for assimilating noisy observations for different noise models. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

5.2.4. Partial Observability

The practicality of DA lies in its ability to assimilate observations that partially or even indirectly characterize the evolution of state variables within a dynamical system. This is particularly valuable when the full system state cannot be directly observed, such as in real-world climate and weather applications. To examine this setting, an RL agent was trained to assimilate noisy observations of select state variables–specifically, the x-variable alone, the x- and y-variables, and the x- and z-variables. The final row of Figure 13 portrays the evolution of RMSE of the aforementioned experiments. The curves demonstrate that, in all cases, the RL agent provides a suitable correction that adequately guides the evolution of the full state. It is noteworthy that the RMSE of the solution obtained using a single RL agent is comparable to, albeit slightly higher than that of the EnKF with an ensemble of 50 realizations. As observed in previous experiments, the averaged RL solution exhibits a lower average RMSE compared to the EnKF. To provide a tangible illustration of the assimilated solution's behavior, the final row of Figure 14 presents curves depicting the temporal evolution of the z-variable for the case with partial system states observability. These plots depict that the RL assimilated solution generally tracks the reference, with occasional discrepancies that typically occur at the peaks and troughs, as expected.

6. Discussion

This paper introduces RL as a novel approach for learning DA corrections. Through extensive experimentation on the Lorenz 63 and 96 dynamical system across various scenarios, we showcase the potential of the proposed approach. Our investigation encompasses both deterministic and stochastic settings, where RL agents are adeptly

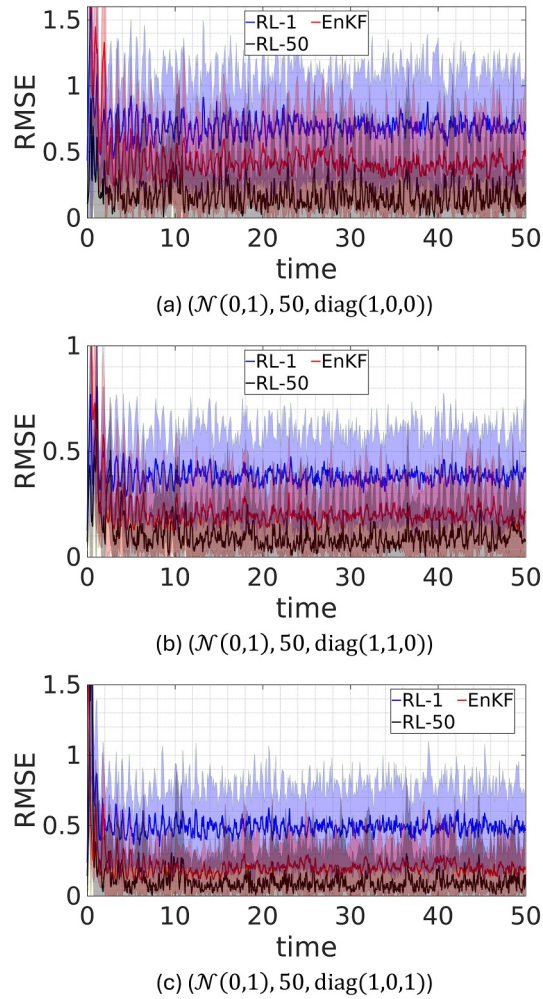


Figure 13. Evolution of the mean RMSE (solid lines) and its $\pm\sigma$ (shadowed) based on 50 experiment repetitions with the Lorenz 63 model. Plotted are the EnKF and RL results for assimilating noisy observations for different observed variables. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

trained to track reference solutions and assimilate noisy data under varying conditions of assimilation window lengths, observational noise distributions, noise levels, and observed state variables.

The proposed RL-DA framework offers a paradigm shift by introducing new degrees of freedom to forecast-correction schemes, allowing for a nonlinear update that satisfies a predefined optimal criteria, such as minimizing the root-mean-squared error in this study, hence, facilitating the discovery of novel correction strategies that are informed by the dynamical system through agent-environment interaction experiences. Furthermore, RL imparts robustness to correction strategies, rendering them stable even in the presence of noisy perturbations and compounding errors. In this work, the RL agent minimizes the ℓ_2 norm of the innovation term, a formulation demonstrated to be equivalent to maximizing the mutual information between observed state variables and their forecast counterparts. Notably, this framework eliminates the need for a reference database as opposed to supervised learning approaches, which are commonly established through the assimilation of noisy observational data using methods such as the EnKF or variational methods (Talagrand & Courtier, 1987).

However, incorporating RL into DA raises further questions warranting further exploration. While we employed the negative of the ℓ_2 norm of the innovation term as the reward function in this study, more sophisticated functions considering system dynamics or ensemble information could potentially enhance the RL

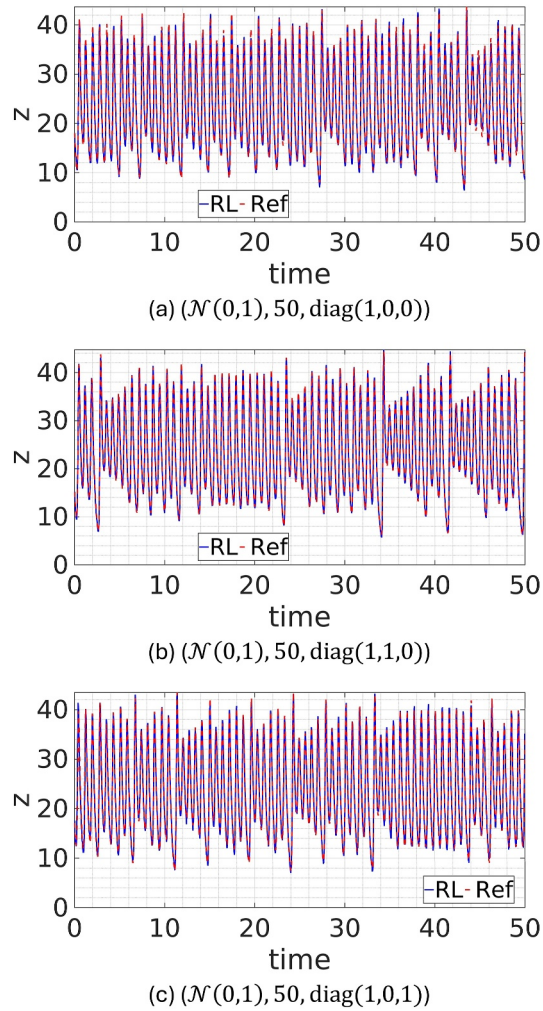


Figure 14. Evolution of the z-variable for a sample RL solution (solid blue lines) and corresponding reference (dashed red line) for the Lorenz 63 system. Plotted are the EnKF and RL results for assimilating noisy observations for different observed variables. The captions beneath each subplot describes the experimental condition in the order of noise distribution, $\mathcal{T}$ the assimilation window size and $\mathcal{H}$ the observational operator.

agent's performance. In addition, the current approach does not exploit statistical information about the ensemble members, but rather employs random sampling to populate the distribution of predicted analyses. However, a more sophisticated approach that uses this information is hypothesized to enhance the RL results. Furthermore, techniques from DA, such as covariance inflation and localization could be employed in the DA-RL strategy to further improve the RL results. Moreover, since the RL agent is trained using the system of differential equations describing the evolution of a dynamical system, we speculate that this would force the agent to adapt and overcome model errors, when present. An overarching concern pertains to the physical validity of RL-derived solutions, which remains an open, fundamental question as is the case with other data-driven approaches when applied to physics-based applications. Although we did not directly encounter violations of physical constraints in our present setup, this avenue remains unexplored and in need for further exploration. Finally, it is worth pointing out that the current framework involves Monte Carlo random sampling to estimate uncertainties, which is not computationally efficient and is prone to sampling errors. This may be mitigated by means of a more sophisticated loss function that takes into account the ensemble statistics, and shall be explored in future work.

Data Availability Statement

All software and data used in the study will be made available upon acceptance at https://github.com/mhammoud115/DA-RL (Hammoud et al., 2024a) and https://zenodo.org/doi/10.5281/zenodo.11186844 (Hammoud et al., 2024b).

Acknowledgments
Research reported in this publication was supported by the Office of Sponsored Research (OSR) at King Abdullah University of Science and Technology (KAUST) CRG Award CRG2020-4336 and Virtual Red Sea Initiative Award REP/1/3268-01-01. The work of E.S.T. was supported in part by NPRP Grant S-0207-200290 from the Qatar National Research Fund (a member of Qatar Foundation).

References

Albrecht, S. V., Christianos, F., & Schäfer, L. (2023). *Multi-agent reinforcement learning: Foundations and modern approaches*. MIT Press. Retrieved from https://www.marl-book.com

Anderson, J. L. (2001). An ensemble adjustment Kalman filter for data assimilation. *Monthly Weather Review*, *129*(12), 2884–2903. https://doi.org/10.1175/1520-0493(2001)129⟨2884:aeakff⟩2.0.co;2

Azouani, A., & Titi, E. S. (2014). Feedback control of nonlinear dissipative systems by finite determining parameters—A reaction-diffusion paradigm. *Evolution Equations and Control Theory*, *3*(4), 579–594. https://doi.org/10.3934/eect.2014.3.579

Bach, E., & Ghil, M. (2023). A multi-model ensemble Kalman filter for data assimilation and forecasting. *Journal of Advances in Modeling Earth Systems*, *15*(1). https://doi.org/10.1029/2022ms003123

Bae, H. J., & Koumoutsakos, P. (2022). Scientific multi-agent reinforcement learning for wall-models of turbulent flows. *Nature Communications*, *13*(1), 1443. https://doi.org/10.1038/s41467-022-28957-7

Bertsekas, D. (2019). *Reinforcement learning and optimal control*. Athena Scientific.

Buizza, C., Quilodrán Casas, C., Nadler, P., Mack, J., Marrone, S., Titus, Z., et al. (2022). Data learning: Integrating data assimilation and machine learning. *Journal of Computational Science*, *58*, 101525. https://doi.org/10.1016/j.jocs.2021.101525

Chen, T., & Chen, H. (1995). Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems. *IEEE Transactions on Neural Networks*, *6*(4), 911–917. https://doi.org/10.1109/72.392253

Eckmann, J. P., & Ruelle, D. (1985). Ergodic theory of chaos and strange attractors. *Reviews of Modern Physics*, *57*(3), 617–656. https://doi.org/10.1103/RevModPhys.57.617

Evensen, G. (2003). The ensemble Kalman filter: Theoretical formulation and practical implementation. *Ocean Dynamics*, *53*(4), 343–367. https://doi.org/10.1007/s10236-003-0036-9

Farchi, A., Laloyaux, P., Bonavita, M., & Bocquet, M. (2021). Using machine learning to correct model error in data assimilation and forecast applications. *Quarterly Journal of the Royal Meteorological Society*, *147*(739), 3067–3084. https://doi.org/10.1002/qj.4116

Foias, C., Jolly, M. S., Kukavica, I., & Titi, E. S. (2001). The Lorenz equation as a metaphor for the Navier-Stokes equations. *Discrete and Continuous Dynamical Systems*, *7*(2), 403–429. https://doi.org/10.3934/dcds.2001.7.403

Ghil, M., & Malanotte-Rizzoli, P. (1991). Data assimilation in meteorology and oceanography. In R. Dmowska, & B. Saltzman (Eds.), *Advances in geophysics* (Vol. 33, pp. 141–266). Elsevier. https://doi.org/10.1016/S0065-2687(08)60442-2

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In Y. W. Teh, & M. Titterington (Eds.), *Proceedings of the thirteenth international conference on artificial intelligence and statistics* (Vol. 9, pp. 249–256). PMLR. Retrieved from https://proceedings.mlr.press/v9/glorot10a.html

Guo, D., Shamai, S., & Verdu, S. (2005). Mutual information and minimum mean-square error in Gaussian channels. *IEEE Transactions on Information Theory*, *51*(4), 1261–1282. https://doi.org/10.1109/TIT.2005.844072

Hammoud, M. A. E. R., Alwassel, H., Ghanem, B., Knio, O., & Hoteit, I. (2023). Physics-informed deep neural network for backward-in-time prediction: Application to Rayleigh–Bénard convection. *Artificial Intelligence for the Earth Systems*, *2*(1), e220076. https://doi.org/10.1175/AIES-D-22-0076.1

Hammoud, M. A. E. R., Raboudi, N., Titi, E. S., Hoteit, I., & Knio, O. (2024a). Data assimilation in chaotic systems using deep reinforcement learning [Software]. https://github.com/mhammoud115/DA-RL

Hammoud, M. A. E. R., Raboudi, N., Titi, E. S., Hoteit, I., & Knio, O. (2024b). Data assimilation in chaotic systems using deep reinforcement learning [Software]. https://zenodo.org/doi/10.5281/zenodo.11186844

Hammoud, M. A. E. R., Titi, E. S., Hoteit, I., & Knio, O. (2022). CDAnet: A physics-informed deep neural network for downscaling fluid flows. *Journal of Advances in Modeling Earth Systems*, *14*(12), e2022MS003051. https://doi.org/10.1029/2022MS003051

Hayden, K., Olson, E., & Titi, E. S. (2011). Discrete data assimilation in the Lorenz and 2d Navier–Stokes equations. *Physica D: Nonlinear Phenomena*, *240*(18), 1416–1425. https://doi.org/10.1016/j.physd.2011.04.021

Hoteit, I., Luo, X., Bocquet, M., Kohl, A., & Ait-El-Fquih, B. (2018). Data assimilation in oceanography: Current status and new directions. *New frontiers in operational oceanography*, 465–512.

Hoteit, I., Pham, D.-T., Triantafyllou, G., & Korres, G. (2008). A new approximate solution of the optimal nonlinear filter for data assimilation in meteorology and oceanography. *Monthly Weather Review*, *136*(1), 317–334. https://doi.org/10.1175/2007MWR1927.1

Houtekamer, P. L., & Mitchell, H. L. (1998). Data assimilation using an ensemble Kalman filter technique. *Monthly Weather Review*, *126*(3), 796–811. https://doi.org/10.1175/1520-0493(1998)126⟨0796:DAUAEK⟩2.0.CO;2

Howard, L. J., Subramanian, A., & Hoteit, I. (2024). A machine learning augmented data assimilation method for high-resolution observations. *Journal of Advances in Modeling Earth Systems*, *16*(1), e2023MS003774. https://doi.org/10.1029/2023MS003774

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., et al. (2021). Highly accurate protein structure prediction with AlphaFold. *Nature*, *596*(7873), 583–589. https://doi.org/10.1038/s41586-021-03819-2

Kalantarov, V. K., & Titi, E. S. (2018). Global stabilization of the Navier-Stokes-Voight and the damped nonlinear wave equations by finite number of feedback controllers. *Discrete and Continuous Dynamical Systems—B*, *23*(3), 1325–1345. https://doi.org/10.3934/dcdsb.2018153

Kalnay, E. (2002). *Atmospheric modeling, data assimilation and predictability*. Cambridge University Press. https://doi.org/10.1017/CBO9780511802270

Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., & Yang, L. (2021). Physics-informed machine learning. *Nature Reviews Physics*, *3*(6), 422–440. https://doi.org/10.1038/s42254-021-00314-5

Kingma, D. P., & Ba, J. (2017). Adam: A method for stochastic optimization.

Kober, J., Bagnell, J. A., & Peters, J. (2013). Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, *32*(11), 1238–1274. https://doi.org/10.1177/0278364913495721

Le Dimet, F.-X., & Talagrand, O. (1986). Variational algorithms for analysis and assimilation of meteorological observations: Theoretical aspects. *Tellus A: Dynamic Meteorology and Oceanography*, *38*(2), 97–110. https://doi.org/10.1111/j.1600-0870.1986.tb00459.x

Lermusiaux, P. F. (2007). Adaptive modeling, adaptive data assimilation and adaptive sampling. *Physica D: Nonlinear Phenomena*, *230*(1), 172–196. https://doi.org/10.1016/j.physd.2007.02.014

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., et al. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Lorenc, A. C. (2003). The potential of the ensemble Kalman filter for NWP—A comparison with 4D-Var. *Quarterly Journal of the Royal Meteorological Society*, *129*(595), 3183–3203. https://doi.org/10.1256/qj.02.132

Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, *20*(2), 130–141. https://doi.org/10.1175/1520-0469(1963)020⟨0130:DNF⟩2.0.CO;2

Lorenz, E. N. (1996). Predictability: A problem partly solved. In *Proceeding of seminar on predictability* (Vol. 1).

Luo, X., & Hoteit, I. (2011). Robust ensemble filtering and its relation to covariance inflation in the ensemble Kalman filter. *Monthly Weather Review*, *139*(12), 3938–3953. https://doi.org/10.1175/MWR-D-10-05068.1

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., et al. (2016). Asynchronous methods for deep reinforcement learning. In M. F. Balcan, & K. Q. Weinberger (Eds.), *Proceedings of the 33rd international conference on machine learning* (Vol. 48, pp. 1928–1937). PMLR. Retrieved from https://proceedings.mlr.press/v48/mniha16.html

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529–533. https://doi.org/10.1038/nature14236

Novati, G., de Laroussilhe, H. L., & Koumoutsakos, P. (2021). Automating turbulence modelling by multi-agent reinforcement learning. *Nature Machine Intelligence*, *3*(1), 87–96. https://doi.org/10.1038/s42256-020-00272-0

Novati, G., & Koumoutsakos, P. (2019). Remember and forget for experience replay. In *Proceedings of the 36th international conference on machine learning* (pp. 1–10).

Ott, E., Hunt, B. R., Szunyogh, I., Zimin, A. V., Kostelich, E. J., Corazza, M., et al. (2004). A local ensemble Kalman filter for atmospheric data assimilation. *Tellus A: Dynamic Meteorology and Oceanography*, *56*(5), 415–428. https://doi.org/10.3402/tellusa.v56i5.14462

Pedatella, N. M., Raeder, K., Anderson, J. L., & Liu, H.-L. (2014). Ensemble data assimilation in the whole atmosphere community climate model. *Journal of Geophysical Research: Atmospheres*, *119*(16), 9793–9809. https://doi.org/10.1002/2014JD021776

Rabier, F. (2005). Overview of global data assimilation developments in numerical weather-prediction centres. *Quarterly Journal of the Royal Meteorological Society*, *131*(613), 3215–3233. https://doi.org/10.1256/qj.05.129

Raboudi, N. F., Ait-El-Fquih, B., & Hoteit, I. (2023). Online estimation of colored observation-noise parameters within an ensemble Kalman filtering framework. *Quarterly Journal of the Royal Meteorological Society*, *149*(754), 1833–1855. https://doi.org/10.1002/qj.4484

Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., & Dormann, N. (2021). Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, *22*(1), 12348–12355.

Recht, B. (2019). A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, *2*(1), 253–279. https://doi.org/10.1146/annurev-control-053018-023825

Rodriguez, E. G. (2021). On disentanglement and mutual information in semi-supervised variational auto-encoders. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 1257–1262).

Sallab, A. E., Abdou, M., Perot, E., & Yogamani, S. (2017). Deep reinforcement learning framework for autonomous driving. *Electronic Imaging*, *29*(19), 70–76. https://doi.org/10.2352/issn.2470-1173.2017.19.avm-023

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms.

Seidler, J. (1971). Bounds on the mean-square error and the quality of domain decisions based on mutual information. *IEEE Transactions on Information Theory*, *17*(6), 655–665. https://doi.org/10.1109/TIT.1971.1054717

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., & Riedmiller, M. (2014). Deterministic policy gradient algorithms. In E. P. Xing, & T. Jebara (Eds.), *Proceedings of the 31st international conference on machine learning* (Vol. 32, pp. 387–395). PMLR. Retrieved from https://proceedings.mlr.press/v32/silver14.html

Subramanian, A. C., Hoteit, I., Cornuelle, B., Miller, A. J., & Song, H. (2012). Linear versus nonlinear filtering with scale-selective corrections for balanced dynamics in a simple atmospheric model. *Journal of the Atmospheric Sciences*, *69*(11), 3405–3419. https://doi.org/10.1175/jas-d-11-0332.1

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning, an introduction*. Bradford Books.

Talagrand, O., & Courtier, P. (1987). Variational assimilation of meteorological observations with the adjoint vorticity equation. I: Theory. *Quarterly Journal of the Royal Meteorological Society*, *113*(478), 1311–1328. https://doi.org/10.1002/qj.49711347812

van Leeuwen, P. J. (2009). Particle filtering in geophysical systems. *Monthly Weather Review*, *137*(12), 4089–4114. https://doi.org/10.1175/2009MWR2835.1

Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., et al. (2019). Grandmaster level in StarCraft ii using multi-agent reinforcement learning. *Nature*, *575*(7782), 350–354. https://doi.org/10.1038/s41586-019-1724-z