



1 **Bridging Data Assimilation and Control: Ensemble Model Predictive**
2 **Control for High-Dimensional Nonlinear Systems**

3 Kenta Kurosawa^a, Atsushi Okazaki^{a,b}, Fumitoshi Kawasaki^c, Shunji Kotsuki^{a,b,d}

4 ^a *Center for Environmental Remote Sensing, Chiba University, Chiba, Japan*

5 ^b *Institute for Advanced Academic Research, Chiba University, Chiba, Japan*

6 ^c *Graduate School of Science and Engineering, Chiba University, Chiba, Japan*

7 ^d *Research Institute of Disaster Medicine, Chiba University, Chiba, Japan*



9 ABSTRACT: Model predictive control (MPC) is an optimization-based control framework for
10 linear and nonlinear systems. MPC estimates control inputs by iterative optimization of a cost func-
11 tion that minimizes deviations from a desired state while accounting for control costs over a finite
12 prediction horizon. This process typically involves direct computations in state space through full
13 model evaluations, making it computationally expensive for high-dimensional nonlinear systems.
14 This study introduces ensemble model predictive control (EnMPC), a novel framework for nonlinear
15 control that combines MPC and ensemble data assimilation. EnMPC directly solves the MPC cost
16 function using ensemble smoother methods, including the four-dimensional ensemble variational
17 assimilation method, ensemble Kalman smoother, and particle smoother. By assimilating pseudo-
18 observations that incorporate information about reference trajectories and constraints, EnMPC
19 mitigates nonlinearity and uncertainty in high-dimensional systems, outperforming conventional
20 MPC in computational efficiency through ensemble approximations. In addition, EnMPC is able
21 to determine optimal weights for control inputs by using the analysis error covariance derived from
22 ensemble data assimilation. We present two different approaches for defining control objectives.
23 The penalty term approach applies penalties when model predictions violate pre-defined constraints
24 by assimilating constraint information as pseudo-observations. In contrast, the trajectory tracking
25 approach assimilates pseudo-observations derived from a reference trajectory to lead the system
26 in the direction of the desired state. We perform numerical experiments with idealized models
27 that capture the chaotic nature of atmospheric systems to show that EnMPC efficiently controls the
28 system and offers flexibility for a variety of control objectives.



29 1. Introduction

30 The intensification of extreme weather events induced by global warming is causing significant
31 damage to human life and property worldwide. As the IPCC sixth assessment report points
32 out, rising temperatures increase the threat by increasing the frequency of heat waves, heavy
33 rains and floods, and the intensity of hurricanes and typhoons (IPCC 2021). The demand for
34 new technological advances is growing as it becomes more difficult to manage the increasing
35 number of extreme weather events with only infrastructure improvements. Since the mid-20th
36 century, researchers have considered interventions such as cloud seeding, where they use silver
37 iodide to induce rainfall. However, while scientific studies have provided evidence to support the
38 effectiveness of the approach to some extent (Langmuir 1948; Ryan and King 1997; Silverman
39 2001), its efficiency and optimization remain areas of active research.

40 Model predictive control (MPC) is a powerful control technique that uses dynamic models to
41 predict future behavior and optimize control actions over a finite time horizon (Morari and Lee
42 1999; Rockett and Hathway 2017; Babu et al. 2019; Schwenzer et al. 2021). As computational
43 power has advanced, the range of its applications has expanded, and new challenges, such as
44 weather control, have become increasingly realistic. However, meteorological systems are highly
45 complex, consisting of numerous interconnected elements such as the atmosphere, oceans, land,
46 and biosphere (Lea et al. 2015; Sluka et al. 2016; Kurosawa et al. 2023). As its behavior exhibits
47 significant nonlinearities, small variations can have unpredictable effects on the entire system
48 (Slingo and Palmer 2011), and the system responds slowly to interventions (Leith 1974), making
49 accurate predictions and control difficult. Moreover, weather models often require significant
50 computational resources due to their high dimensionality and the need for fine temporal and spatial
51 resolutions. Given these characteristics of weather systems, proper handling of uncertainty and
52 the heavy computational cost of calculating optimal control inputs are key challenges for achieving
53 effective weather control.

54 To properly handle uncertainty, data assimilation integrates observations and numerical models
55 to more accurately estimate the state of the system, and it is widely used in weather forecast-
56 ing (Houtekamer and Mitchell 1998; Kalnay 2003; Leutbecher and Palmer 2008; Evensen 2009).
57 Miyoshi and Sun (2022) proposed a new experimental framework to systematically evaluate control
58 approaches through ensemble prediction. In the framework, known as the control simulation exper-



59 iment (CSE), they used ensemble data assimilation for state estimation. Subsequently, Kawasaki
60 and Kotsuki (2024) integrated a conventional MPC method and achieved efficient control with
61 minimal input within the CSE framework. However, the computational cost of calculating optimal
62 control inputs remains high, and there is a need to develop more efficient control methods.

63 Sawada (2024a,b) proposed a weather control method that combines ensemble data assimilation
64 and MPC, utilizing the ensemble Kalman filter (EnKF) and ensemble Kalman smoother (EnKS) to
65 solve the MPC problem efficiently. Traditional MPC requires direct computations in state spaces
66 and explicit calculation of system evolution within the prediction horizon, whereas ensemble
67 approximations use statistical representations, enabling more efficient control of complex systems.
68 The EnKF-based control method, which directly utilizes the existing EnKF architecture, offers
69 flexibility for geoscience applications but still faces several challenges. First, when calculating
70 the optimal control inputs, the system's behavior within the evaluation horizon or window of the
71 cost function is assumed to be approximately linear. In systems with strong nonlinearity, this
72 approximation does not hold, and errors are likely to occur when calculating the optimal control
73 input (Zhang et al. 2009; Kurosawa and Poterjoy 2021). Second, as used in Sawada (2024a), many
74 control problems commonly add penalty terms to the cost function to handle constraint violations
75 in control objectives. In the penalty-based approaches, when control objectives are unclear or
76 multiple objectives must be balanced, designing the cost function and setting penalties becomes
77 challenging, potentially reducing performance and causing unintended behavior.

78 To address these challenges, the current study extends the methodology of using ensemble data
79 assimilation for solving MPC problems, building upon the insights of Sawada (2024a). Specifically,
80 we propose an ensemble model predictive control (EnMPC) framework that employs various
81 ensemble data assimilation techniques, including 4D-ensemble-Var (4DEnVar), particle filter (PF),
82 and particle smoother (PS). This approach expands the range of tools available for solving MPC
83 problems in high-dimensional nonlinear systems. As part of this framework, the EnMPC includes
84 the method proposed by Sawada (2024a), which uses the EnKF and EnKS to solve MPC problems.
85 Furthermore, the EnMPC framework introduces not only the penalty-based approach but also a
86 trajectory-tracking approach to achieve control, providing greater flexibility in addressing diverse
87 control objectives. To demonstrate the effectiveness of the proposed EnMPC framework, we
88 conduct a comparison with conventional MPC approaches.



89 The manuscript is organized in the following manner. Section 2 provides a brief overview of
 90 ensemble data assimilation and MPC. We introduce EnMPC in Sec. 3, and Sec. 4 describes the
 91 experimental setup. Section 5 presents the experimental results, and the last section concludes the
 92 paper with a summary of the key findings, potential applications, and directions for future research.

93 **2. MPC and data assimilation**

94 This section provides a brief overview of MPC and ensemble data assimilation, which constitute
 95 the proposed EnMPC framework. We begin by presenting the MPC algorithm for dealing with
 96 control problems. Subsequently, we outline ensemble data assimilation, focusing on 4DVar,
 97 EnKF, and PF. This section explains MPC and data assimilation individually, while Sec. 3
 98 highlights their similarities, differences, and how they are combined to form EnMPC.

99 *a. MPC*

100 MPC is a control strategy that optimizes control inputs by using a dynamic model to predict the
 101 future behavior of the system. MPC solves an optimization problem at each time step to minimize
 102 a cost function over a finite predictive horizon. The specific design of the cost function depends
 103 on the application, but the general formulation can be expressed as:

$$J(\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_{T_c}) = \underbrace{\sum_{t=0}^{T_c} \mathbf{u}_t^\top \mathbf{C}^u \mathbf{u}_t}_{J_{\text{input}}} + \underbrace{\sum_{t=0}^{T_p} (\mathbf{r}_t - H^c(\mathbf{x}_t))^\top \mathbf{C}^{r-1} (\mathbf{r}_t - H^c(\mathbf{x}_t))}_{J_{\text{state}}}. \quad (1)$$

s.t. $\mathbf{x}_{t+1} = M_t(\mathbf{x}_t, \mathbf{u}_t).$

104 Here, \mathbf{x}_t denotes the state variable at time t . The next state \mathbf{x}_{t+1} is obtained by integrating the
 105 nonlinear forecast model operator M_t forward from the current state \mathbf{x}_t and the control input \mathbf{u}_t .
 106 The control input cost J_{input} is typically optimized over a shorter control horizon T_c within the
 107 prediction horizon T_p . J_{input} penalizes the magnitude of the control input, preventing it from being
 108 excessively large. The state deviation cost J_{state} evaluates the difference between model-predicted
 109 states and the control objective \mathbf{r} , and the optimization problem is performed over a finite prediction
 110 horizon T_p . H^c is an operator that maps the state variables \mathbf{x} to the control variables. \mathbf{C}^u and



111 \mathbf{C}^r are weighting matrices for the control input \mathbf{u} and the deviations between state variables and
 112 control objective, respectively.

113 Among the two components of the cost function in (1), the state deviation cost J_{state} typically has
 114 the highest computational cost. This is because it involves predicting and evaluating the future states
 115 of the system over the entire prediction horizon, which requires extensive computations, especially
 116 for complex or nonlinear systems. The ensemble approximation can mitigate this computational
 117 cost by using representative trajectories to approximate future states, as discussed in Sec. 3.

118 *b. The four-dimensional variational method (4DVar) and 4DEnVar*

119 The 4DVar method estimates the optimal initial state \mathbf{x}_0 over a time window by considering the
 120 misfits between observations and forecast model states at multiple times. This process is achieved
 121 by minimizing the following cost function (Talagrand 2014; Bannister 2017):

$$J(\mathbf{x}_0) = \underbrace{(\mathbf{x}_0 - \mathbf{x}_0^b)^\top \mathbf{B}^{-1} (\mathbf{x}_0 - \mathbf{x}_0^b)}_{J_{\text{background}}} + \underbrace{\sum_{t=0}^{\tau} (\mathbf{y}_t - H(\mathbf{x}_t))^\top \mathbf{R}^{-1} (\mathbf{y}_t - H(\mathbf{x}_t))}_{J_{\text{observation}}}, \quad (2)$$

s.t. $\mathbf{x}_{t+1} = M_t(\mathbf{x}_t)$

122 The first term in (2) qualifies the difference between the initial guess (background or prior) \mathbf{x}_0^b and the
 123 estimated state \mathbf{x}_0 , weighted by the background error covariance matrix \mathbf{B} . The second term in (2)
 124 measures the misfit between the state variables and the observations \mathbf{y} at times $t = 0, 1, 2, \dots, \tau$. The
 125 observation operator H maps the state \mathbf{x} to the observation space, and \mathbf{R} represents the observation
 126 error covariance matrix. The time window τ is referred to as the data assimilation window and
 127 plays the same role as the prediction horizon T_p in MPC. Therefore, the second term $J_{\text{observation}}$ in
 128 (2) serves a similar purpose to the state deviation cost J_{state} in the MPC cost function (1), as both
 129 evaluate the discrepancies between the predicted states and the target values or observations over
 130 a specific time horizon.

131 Operational systems often implement 4DVar using an incremental approach to utilize the lin-
 132 earized model instead of the full nonlinear model (Courtier et al. 1994). Defining $\delta \mathbf{x}_0 = \mathbf{x}_0 - \mathbf{x}_0^b$,



the cost function $J(\mathbf{x}_0)$ in (2) as becomes:

$$J(\delta\mathbf{x}_0) = \underbrace{\delta\mathbf{x}_0^\top \mathbf{B}^{-1} \delta\mathbf{x}_0}_{J_{\text{background}}} + \underbrace{\sum_{t=0}^{\tau} (\mathbf{H}\delta\mathbf{x}_t - \mathbf{d}_t)^\top \mathbf{R}^{-1} (\mathbf{H}\delta\mathbf{x}_t - \mathbf{d}_t)}_{J_{\text{observation}}}, \quad (3)$$

s.t. $\delta\mathbf{x}_{t+1} = \mathbf{M}_t(\delta\mathbf{x}_t)$

where \mathbf{M}_t and \mathbf{H} are the tangent linear operators of M_t and H , respectively. The innovation vector \mathbf{d}_t is defined as $\mathbf{d}_t = \mathbf{y}_t - H[M_t(\mathbf{x}_0^b)]$.

The convergence rate of the optimization problem depends on the condition number of the Hessian matrix (Zupanski 1996). In operational data assimilation systems using atmospheric models, the dimension of the state vector is typically on the order of $O(10^{10})$ or greater. This results in a background error covariance matrix \mathbf{B} that is too large to be explicitly represented or handle directly. To address this computational challenge, operational systems commonly employ the following approach (Buehner 2005; Wang et al. 2010; Zhu et al. 2022):

$$\begin{cases} \delta\mathbf{x}_0 = \mathbf{U}^x \mathbf{v}, \\ \mathbf{H}\delta\mathbf{x}_t = \mathbf{U}_t^y \mathbf{v}, \end{cases} \quad (4)$$

Here, \mathbf{U}^x is a square root of the background error covariance matrix ($\mathbf{B} = \mathbf{U}^x \mathbf{U}^{x\top}$; Lorenc 2003), and \mathbf{v} is the new control variable in the reduced-dimension space. The initial perturbation $\delta\mathbf{x}_0$ and the observation perturbation $\mathbf{H}\delta\mathbf{x}_t$ are projected onto a subspace spanned by ensemble members using the transformation matrices \mathbf{U}^x and \mathbf{U}_t^y , respectively. The perturbation matrices \mathbf{U}^x and \mathbf{U}^y are defined as follows:

$$\begin{cases} \mathbf{U}^x = \frac{1}{\sqrt{N_e-1}} \begin{bmatrix} \delta\mathbf{x}^{(1)} & \delta\mathbf{x}^{(2)} & \dots & \delta\mathbf{x}^{(N_e)} \end{bmatrix} \\ \mathbf{U}^y = \frac{1}{\sqrt{N_e-1}} \begin{bmatrix} \delta\mathbf{y}^{(1)} & \delta\mathbf{y}^{(2)} & \dots & \delta\mathbf{y}^{(N_e)} \end{bmatrix}, \end{cases} \quad (5)$$

where N_e is the ensemble size, $\delta\mathbf{x}^{(k)}$ and $\delta\mathbf{y}^{(k)}$ are the k -th ensemble perturbations for the model state and observation space, respectively. Perturbations in observation space are calculated using the tangent linear observation operator, where $\delta\mathbf{y} = \mathbf{H}\delta\mathbf{x}$. By adopting this transformation, the cost



function is reformulated as:

$$J(\mathbf{v}) = \underbrace{\mathbf{v}^\top \mathbf{v}}_{J_{\text{background}}} + \underbrace{\sum_{t=0}^{\tau} (\mathbf{U}_t^y \mathbf{v} - \mathbf{d}_t)^\top \mathbf{R}^{-1} (\mathbf{U}_t^y \mathbf{v} - \mathbf{d}_t)}_{J_{\text{observation}}}. \quad (6)$$

To minimize (6), \mathbf{v} must satisfy the condition $(\partial J / \partial \mathbf{v})^\top = \mathbf{0}$. As a result, this approach eliminates the need for an adjoint model, as all calculations occur within the subspace spanned by the ensemble samples. This incremental 4DVar approach combines with ensemble-based transformations thus balances computational efficiency and the practical constraints of high-dimensional data assimilation systems. For further details on these methods, we encourage readers to review the mathematical descriptions in Liu et al. (2009), Fairbairn et al. (2014), Poterjoy and Zhang (2015), and Kurosawa and Poterjoy (2021).

c. EnKF and EnKS

In this study, the control method based on the EnKF adopts the framework proposed in Sawada (2024a). The EnKF minimizes the following cost function to obtain the analysis state:

$$J(\mathbf{x}_0) = \underbrace{(\mathbf{x}_0 - \overline{\mathbf{x}_0^b})^\top \mathbf{P}^{b-1} (\mathbf{x}_0 - \overline{\mathbf{x}_0^b})}_{J_{\text{background}}} + \underbrace{(\mathbf{y}_0 - H(\mathbf{x}_0))^\top \mathbf{R}^{-1} (\mathbf{y}_0 - H(\mathbf{x}_0))}_{J_{\text{observation}}}. \quad (7)$$

Here, $\overline{\mathbf{x}_0^b}$ is the ensemble mean of the background state variables and \mathbf{P}^b represents the background error covariance matrix. As in 4DVar, MPC and EnKF consider similar cost components, taking into account the background information and discrepancies in their respective frameworks.

The EnKF efficiently reduces the computational cost by representing the error covariance matrix \mathbf{P}^b statistically using ensemble members as follows (Evensen 1994; Whitaker and Hamill 2002; Houtekamer and Zhang 2016):

$$\mathbf{P}^b = \mathbf{E} \mathbf{E}^\top, \quad (8)$$

$$\mathbf{E} = \frac{1}{\sqrt{N_e - 1}} [\delta \mathbf{x}^{(1)}, \dots, \delta \mathbf{x}^{(N_e)}], \quad (9)$$



167 where \mathbf{E} is the matrix of ensemble members, with each column representing the perturbation
 168 from the forecast state. Thus, analytically solving the cost function in (7) yields the update of
 169 the ensemble mean. Unlike the variational methods discussed in Sec.2.b, which require iterative
 170 numerical optimization to minimize their respective cost functions, EnKF does not require such
 171 iterations.

172 Regarding the update of ensemble members, we obtain the ensemble perturbation matrix \mathbf{X}^a
 173 using the ensemble transform Kalman filter (ETKF; Bishop et al. 2001; Hunt et al. 2007), as
 174 follows:

$$\mathbf{X}^a = \mathbf{X}^b \mathbf{W}^a, \quad (10)$$

$$\mathbf{W}^a = [(\mathbf{N}_e - 1) \tilde{\mathbf{P}}^a]^{1/2}, \quad (11)$$

$$\tilde{\mathbf{P}}^a = [(\mathbf{N}_e - 1) \mathbf{I} + (\mathbf{Y}^b)^\top \mathbf{R}^{-1} \mathbf{Y}^b]^{-1}. \quad (12)$$

175 Here, \mathbf{X}^b is the background perturbations, and $\tilde{\mathbf{P}}^a$ represents the analysis error covariance matrix
 176 in the transformed space. \mathbf{Y}^b represents the perturbation of the background ensemble in the
 177 observation space, and the weights \mathbf{W}^a are then derived based on the analysis covariance. Similarly
 178 to 4DEnVar, which uses ensemble approximations to project initial and observation perturbations
 179 onto a subspace spanned by ensemble members, the ETKF efficiently reduces the dimensionality
 180 of the analysis problem with ensemble-based transformations.

181 While EnKF is effective for real-time state estimation, EnKS improves estimation accuracy
 182 further by considering observations over a time window and incorporating their influence re-
 183 spectively. In this study, we employ a four-dimensional extension of the ETKF, which uses
 184 temporal correlations within the data to achieve more accurate estimation. For a comprehensive
 185 explanation, please refer to Miyoshi and Aranami (2006) and Hunt et al. (2007).

186 *d. PF and PS*

187 Variational methods and EnKF estimate the analysis state by assuming Gaussian error statistics
 188 for the background and observations and minimizing the cost functions defined in (2) and (7). In
 189 contrast, the PF does not assume Gaussianity or linearity but approximates the entire probability
 190 distribution of the state as a set of particles (ensembles or samples). By assigning a likelihood to



each particle, PF estimates the analysis state, making it suitable for systems with strong nonlinearity and non-Gaussianity. The particle distribution plays a similar role to the error covariance matrices (**B** and **P**) used in the variational methods and EnKF. Unlike these methods, however, PF does not explicitly calculate the error covariance; instead, the particle distribution implicitly represents their statistical properties of the background error covariance. For each particle $\mathbf{x}^{(k)}$, the likelihood is calculated as:

$$p(\mathbf{y}|\mathbf{x}^{(k)}) \propto \exp\left(-\frac{1}{2}(\mathbf{y} - H(\mathbf{x}^{(k)}))^T \mathbf{R}^{-1}(\mathbf{y} - H(\mathbf{x}^{(k)}))\right). \quad (13)$$

This calculation resembles the state deviation term J_{state} in (1) for MPC, where posterior states are penalized based on their deviation from the reference. The likelihoods are normalized to produce the particle weights $\lambda^{(k)}$:

$$\lambda^{(k)} = \frac{p(\mathbf{y}|\mathbf{x}^{(k)})}{\sum_{m=1}^{N_e} p(\mathbf{y}|\mathbf{x}^{(m)})}. \quad (14)$$

Using the weighted particles, PF approximates the posterior distribution (filter distribution) as:

$$p(\mathbf{x}|\mathbf{y}) \approx \sum_{m=1}^{N_e} \lambda^{(m)} \delta(\mathbf{x} - \mathbf{x}^{(m)}), \quad (15)$$

where $\delta(\mathbf{x} - \mathbf{x}^{(k)})$ represents a Dirac delta function centered at particle $\mathbf{x}^{(k)}$. This representation indicates that the posterior distribution is expressed as a discrete set of weighted particles. To better approximate the posterior distribution and mitigate degeneracy, where some particles have negligible weights, a resampling step is performed. During resampling, particles with higher weights are replicated, while those with lower weights are discarded, ensuring the ensemble remains focused on the most likely regions of the state space. This resampling process mirrors the adaptive selection of control inputs in MPC, which focuses optimization efforts on trajectories that minimize the cost function.

The PF is a method for sequentially estimating states, while the PS uses future observation data to provide more accurate state estimates. Applying the weights calculated during the filter update within a data assimilation window, PS uses of the future weights to find the smoother solution at any point throughout the window. This approach is justified by the Markov property, where the system's future evolution depends solely on its current state (Chopin and Papaspiliopoulos 2020;



214 Nyobe et al. 2023). By taking advantage of this feature, the smoother can produce more accurate
215 estimates over the assimilation window by using future data and previously calculated weights.

216 We note that several studies propose strategies to address degeneracy and maintain particle
217 diversity (e.g., Penny and Miyoshi 2016; Potthast et al. 2019; Kotsuki et al. 2022). These differences
218 include the resampling strategy, techniques to mitigate particle collapse, and localization to manage
219 high-dimensional systems. The current study adopts the PF and PS algorithm based on the recently
220 proposed PF by Poterjoy (2022), as it employs regularization and iterative updates to effectively
221 address degeneracy and maintain particle diversity. For more detailed information on this approach,
222 please refer to Poterjoy (2016, 2022) and Kurosawa and Poterjoy (2023).

223 3. Ensemble Model Predictive Control

224 Section 2 provides an overview of conventional MPC and ensemble data assimilation, highlight-
225 ing their shared goal of determining optimal inputs based on the current state and future predictions.
226 This section introduces a new control technique called EnMPC, which integrates these two meth-
227 ods. Since EnMPC uses the principles of data assimilation, it incorporates pseudo-observations
228 that contain information about constraints and reference trajectories typically used in MPC. These
229 pseudo-observations are assimilated in a manner similar to actual observations in data assimila-
230 tion, allowing the cost function in EnMPC to adopt a structure similar to that in ensemble data
231 assimilation.

232 Sawada (2024a) focuses on similarities and differences between EnKF and MPC and introduces
233 EnKF-based EnMPC. Extending this concept, this section focuses on the mathematical formulation
234 of EnMPC, using ideas from 4DVar to develop a 4DVar-based EnMPC approach. We define the
235 formulation of EnMPC in a straightforward manner by modifying the MPC cost function in (1) to
236 make it closer in structure to that of 4DVar in (6).

237 First, data assimilation focuses on state estimation by updating the initial conditions for model
238 integration, while MPC estimates control inputs applied during the control horizon T_c . The
239 proposed EnMPC framework treats the control inputs as acting only at the initial time, similar to
240 how data assimilation updates the initial states. While this assumption simplifies the framework,
241 extending EnMPC to optimize control inputs over the entire control horizon T_c remains an important
242 direction for future research. Second, the conventional MPC uses the reference vector \mathbf{r} in (1) to



represent the desired state. In EnMPC, we reformulate the reference vector as a pseudo-observation vector \mathbf{y}^p . This allows EnMPC to handle reference information in the same way data assimilation incorporates observations. The cost function for EnMPC is therefore expressed as follows:

$$J(\mathbf{x}_0) = \underbrace{(\mathbf{x}_0 - \bar{\mathbf{x}}_0^a)^\top \mathbf{P}^{a-1} (\mathbf{x}_0 - \bar{\mathbf{x}}_0^a)}_{J_{\text{input}}} + \underbrace{\sum_{t=0}^{T_p} (\mathbf{y}_t^p - H^p(\mathbf{x}_t))^\top \mathbf{C}^{r-1} (\mathbf{y}_t^p - H^p(\mathbf{x}_t))}_{J_{\text{state}}}. \quad (16)$$

s.t. $\mathbf{x}_{t+1} = M_t(\mathbf{x}_t)$.

Here, \mathbf{P}^a is the analysis error covariance matrix, as the ensemble updated by data assimilation can be used directly. H^p is the operator that maps the state vector to the pseudo-observation space. As described in Section 2b, applying ensemble approximations to the cost function in (16) yields:

$$J(\mathbf{v}) = \underbrace{\mathbf{v}^\top \mathbf{v}}_{J_{\text{input}}} + \underbrace{\sum_{t=0}^{T_p} (\mathbf{U}_t^y \mathbf{v} - \mathbf{d}_t^p)^\top \mathbf{C}^{r-1} (\mathbf{U}_t^y \mathbf{v} - \mathbf{d}_t^p)}_{J_{\text{state}}}, \quad (17)$$

where the innovation vector \mathbf{d}_t^p is defined as $\mathbf{d}_t^p = \mathbf{y}_t - H^p[M_t(\bar{\mathbf{x}}_0^a)]$. The gradient of the cost function in (17) with respect to \mathbf{v} is expressed as:

$$\left(\frac{\partial J}{\partial \mathbf{v}} \right)^\top = \mathbf{v} + \sum_{t=0}^{T_p} \mathbf{U}_t^{y\top} \mathbf{C}^{r-1} [\mathbf{U}_t^y \mathbf{v} - \mathbf{d}_t^p] \quad (18)$$

This expression shows that solving the EnMPC optimization problem does not require the full nonlinear model or its tangent linear model, as the ensemble approximations are used to calculate the gradient.

A key feature of EnMPC is its ability to assimilate pseudo-observations in a manner similar to actual observations in data assimilation. Therefore, the EnMPC approach, which directly solves the MPC cost function using ensemble estimations, is not limited to 4D-EnVar-based framework, but can also be applied to EnKS- or PS-based frameworks. Moreover, EnMPC offers flexibility in setting control objectives through the use of the pseudo-observations and operators. This study introduces two approaches for defining control objectives. The first, referred to as the "penalty term approach," creates a pseudo-observation vector only when the model prediction exceeds a



predefined threshold, as used in Sawada (2024a). The second, called the "trajectory tracking approach," generates pseudo-observation data directly from the reference trajectory, enabling straightforward objective definition. We provide more details in Sec. 4c. Lastly, EnMPC can appropriately handle sampling errors and uncertainties by incorporating techniques from ensemble data assimilation, such as localization and inflation, as detailed in Sawada (2024b).

4. Experimental settings

In this section, we describe the experimental setup used to evaluate the effectiveness of the proposed EnMPC through numerical experiments using the Lorenz63 (Lorenz 1963) model. Our experiments follow the CSE procedure (Miyoshi and Sun 2022; Sun et al. 2023; Ouyang et al. 2023; Kawasaki and Kotsuki 2024; Sawada 2024a).

a. Experimental procedure

Figure 1 illustrates the process of the CSE using the proposed EnMPC. The procedure consists of the following steps:

1. To obtain an accurate estimate of the current state of the system, we first simulate observations from the nature run (NR; or the true state of the system). We then perform a conventional ensemble data assimilation using these simulated observations, which corresponds to the filter update (Fig. 1a). This step includes estimating unobserved state variables that are targets for control. The outcome of this process provides the initial conditions necessary for the subsequent control step.
2. Based on the state estimated in the previous step, we determine the optimal control input using the proposed EnMPC. We consider two approaches for control input determination:

(a) Penalty term approach

This approach uses a pseudo-observation operator, which acts as a penalty function commonly used in the conventional MPC. Pseudo-observations are generated when the model prediction violates the predefined constraints, effectively penalizing unsuitable behavior (Fig. 1b1).



287 (b) Trajectory tracking approach

288 In the current study, pseudo-observations are directly derived from the reference tra-
 289 jectory, making it straightforward to guide the system toward the desired state (Fig.
 290 1b2).

291 3. The optimal control input determined in the second step is applied to the NR to perform the
 292 control, and the state is integrated forward to the next time step. Similarly, we apply the same
 293 control input to the ensemble members and predict their states for the next time step. With the
 294 updated system state and ensemble predictions, we restart the CSE cycle from the first step
 295 (Fig. 1c).

296 Here, we emphasize that for state estimation in the first step (Fig. 1a), we employ conventional
 297 ensemble data assimilation methods, corresponding to the filter update. In contrast, the second
 298 step (Fig. 1b) utilizes the proposed EnMPC, which is based on an ensemble smoother update, to
 299 determine the optimal control inputs. For data assimilation in the first step, we consistently use the
 300 ETKF, regardless of which ensemble smoother update method (4DVar, EnKS, or PS) is employed
 301 in EnMPC in the second step. This uniformity ensures that any differences in performance are
 302 solely due to the choice of method in EnMPC in the second step and not influenced by variations
 303 in the state estimation in the first step. Lastly, the current study adopts a moving horizon window
 304 of one step. That is, regardless of the length of the prediction horizon used in EnMPC, data
 305 assimilation and control input estimation are performed at every time step in each cycle.

312 *b. Model description*

313 The current study uses the Lorenz63 (Lorenz 1963) model for testing the proposed control
 314 method. Although relatively simple in structure, the model is widely employed as a testbed for
 315 understanding chaotic system behavior. This study aims to demonstrate the effectiveness of EnMPC
 316 for control and parameter estimation in such chaotic systems.

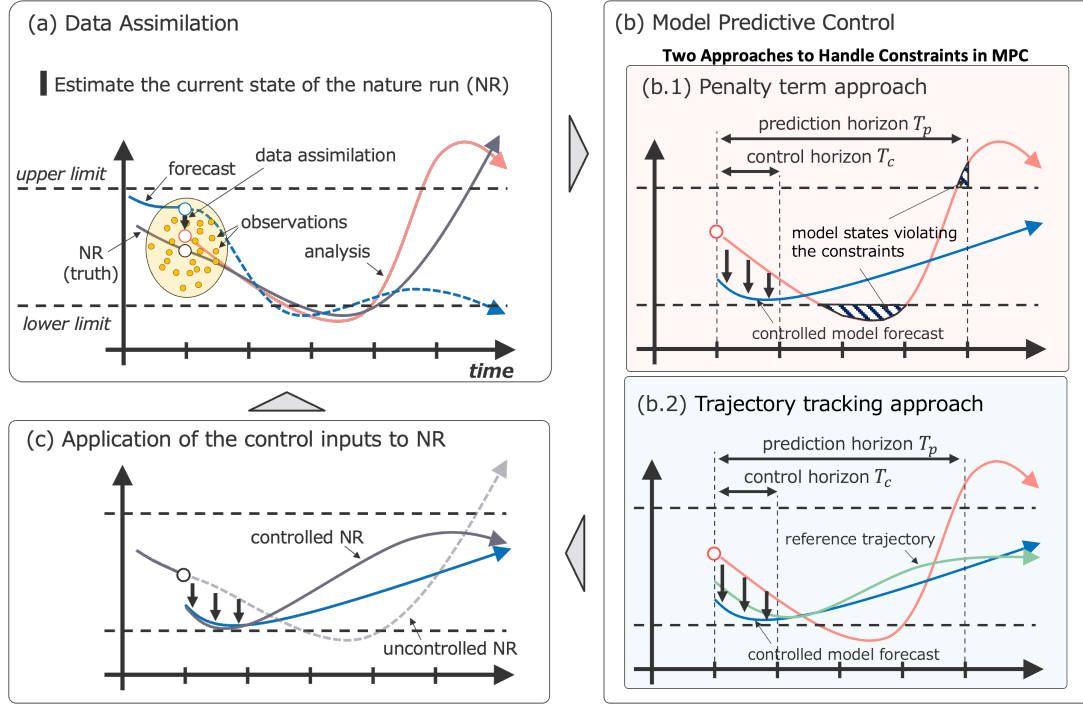


FIG. 1. Algorithmic flow of the proposed EnMPC-based CSE for a system with upper and lower limits. (a) State estimation: estimates the current state of the system using data assimilation (filter update). (b) Control input optimization: determines the optimal control inputs using the proposed EnMPC framework based on ensemble forecasts; (b.1) penalty term approach and (b.2) trajectory tracking approach. (c) Application of control inputs: applies the optimized control inputs to the NR, integrates the system state forward to the next time step, and returns to the filter update step (a), restarting the CSE cycle.

The Lorenz63 model is a simplified model of atmospheric convection and is represented by the following set of ordinary differential equations with three state variables:

$$\begin{cases} \frac{dx}{dt} = \sigma(y - x), \\ \frac{dy}{dt} = x(\rho - z) - y, \\ \frac{dz}{dt} = xy - \beta z. \end{cases} \quad (19)$$



319 Following Lorenz (1963), we set system parameters $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$. The time
 320 step is set to $\Delta t = 0.01$ (units defined arbitrarily as 1 hour; see Lorenz (1963)). The Lorenz63
 321 model is characterized by its chaotic trajectory, which oscillates around two unstable fixed points,
 322 $(\pm\sqrt{72}, \pm\sqrt{72}, 27)^\top$ (Kaiser et al. 2018).

323 Using the Lorenz63 model, the current study investigates two scenarios for control input estima-
 324 tion: estimating only u_x , as shown in (20), and estimating all three control variables u_x , u_y , and
 325 u_z , as shown in (21):

$$\begin{cases} \frac{dx}{dt} = \sigma(y - x) + u_x, \\ \frac{dy}{dt} = x(\rho - z) - y, \\ \frac{dz}{dt} = xy - \beta z, \end{cases} \quad (20)$$

326 and

$$\begin{cases} \frac{dx}{dt} = \sigma(y - x) + u_x, \\ \frac{dy}{dt} = x(\rho - z) - y + u_y, \\ \frac{dz}{dt} = xy - \beta z + u_z. \end{cases} \quad (21)$$

327 The control objective in the current study is to keep the value of x in the model positive, ensuring
 328 that the system avoids undesired negative states. Note that the control inputs are applied to the time
 329 derivatives of the state variables, rather than the states themselves.

330 *c. Pseudo-observations and operators*

331 In the proposed EnMPC framework, we address control problems using two approaches: the
 332 penalty term approach and the trajectory tracking approach. Each approach employs different
 333 methods for generating pseudo-observations \mathbf{y}^p and operators H^p . Throughout our experiments,
 334 we set the pseudo-observation error covariance matrix \mathbf{C}^r , which acts as the weighting matrix for
 335 the deviations between state variables and control objectives, to $\mathbf{C}^r = 0.01\mathbf{I}$, where \mathbf{I} is the identity
 336 matrix. This configuration is based on insights from preliminary experiments and the detailed
 337 investigation in Sawada (2024a,b).



338 1) PENALTY TERM APPROACH

339 In the penalty term approach, we generate pseudo-observations to ensure that variables remain
 340 within specified thresholds. We set the pseudo-observation value to the threshold and assimilate it
 341 into the state space via a pseudo-observation operator. Sawada (2024a) employs a similar strategy,
 342 designing the control operator to impose penalties when constraints are violated. This approach
 343 effectively makes the pseudo-observation operator serve the same role as the penalty function
 344 commonly used in conventional MPC.

345 The control objective of the current study is to keep the x value positive in the Lorenz63 model.
 346 When we apply the penalty term approach for the objective (as detailed in Sec. 5a), we use the
 347 following pseudo-observation operator H^p :

$$H^p(x) = \frac{\log(1 + \exp(-ax))}{a}, \quad (22)$$

348 where a is a positive constant that determines the sharpness of the penalty function. As shown
 349 in Fig. 2, when $a = 100$, the function approximates a hinge function that activates the penalty
 350 only when x becomes less than zero. To keep the value of x non-negative, we set the pseudo-
 351 observation $y^p = 0$. We then use a pseudo-observation operator H^p to project the model state x
 352 into the observation space $H^p(x)$, effectively imposing a penalty when x violates the constraint. A
 353 smaller a results in a smoother transition, applying penalties even when x is above the threshold
 354 but approaching the threshold, as shown in Fig. 2.

355 Figure 3 illustrates the impact of changing the parameter a in the pseudo-observation operator
 356 using the Lorenz63 model. Control input u_x is calculated at each time step using (20), and the
 357 prediction horizon T_p is set to 48 steps (= 48 hour). For this demonstration, we use the 4DEnVar-
 358 based EnMPC with 10 ensemble members. The parameters for this experiment are summarized in
 359 Table 1a.

360 When $a = 100$, the control inputs are relatively large due to delayed activation of the penalty term,
 361 resulting in spike-like control behavior (Fig. 3a,d). Decreasing the value of a activates the penalty
 362 more gradually, allowing the control to respond earlier, thus preventing $x < 0$ more smoothly (Figs.
 363 3b–c and e–f). These results show that the choice of a is critical and depends on the specific control
 364 objectives. When the control objective is to maintain the system state close to the threshold, a



larger a may be necessary, leading to larger and abrupt control inputs. On the other hand, when staying further from the threshold is acceptable, a smaller a can reduce the overall control inputs, although the model states may not closely approach the threshold. This highlights the importance of selecting an appropriate pseudo-observation operator to balance the desired control objectives with the acceptable magnitude of control inputs.

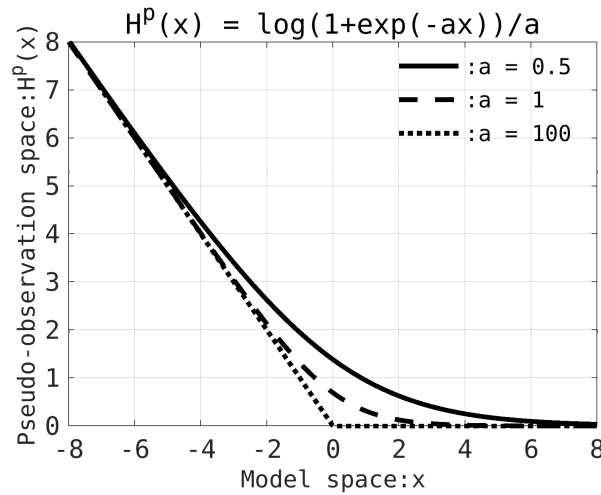


FIG. 2. Comparison of the pseudo-observation operator $H^p(x) = \log(1 + \exp(-ax))/a$ used in this study for different values of the positive constant parameter a . The solid line, dashed line, and dotted line represent the cases where $a = 0.5$, $a = 1$, and $a = 100$, respectively. The horizontal axis represents values in the model space, while the vertical axis represents the values projected into the pseudo-observation space using the operator.

2) TRAJECTORY TRACKING APPROACH

In the trajectory tracking approach, the current study first defines a reference trajectory that satisfies the desired constraints. We then control or guide the system to follow this trajectory by assimilating pseudo-observations. The pseudo-observations are generated by taking the states of the reference trajectory at each observation time.

For the experiment using the Lorenz63 model (as detailed in Sec. 5b), we use the trajectory generated by Kawasaki and Kotsuki (2024) as the reference. This trajectory satisfies the constraint $x > 0$ and is obtained using conventional MPC by applying control inputs u_x , u_y , and u_z to the Lorenz63 model. We generate the pseudo-observations from the reference every time step for

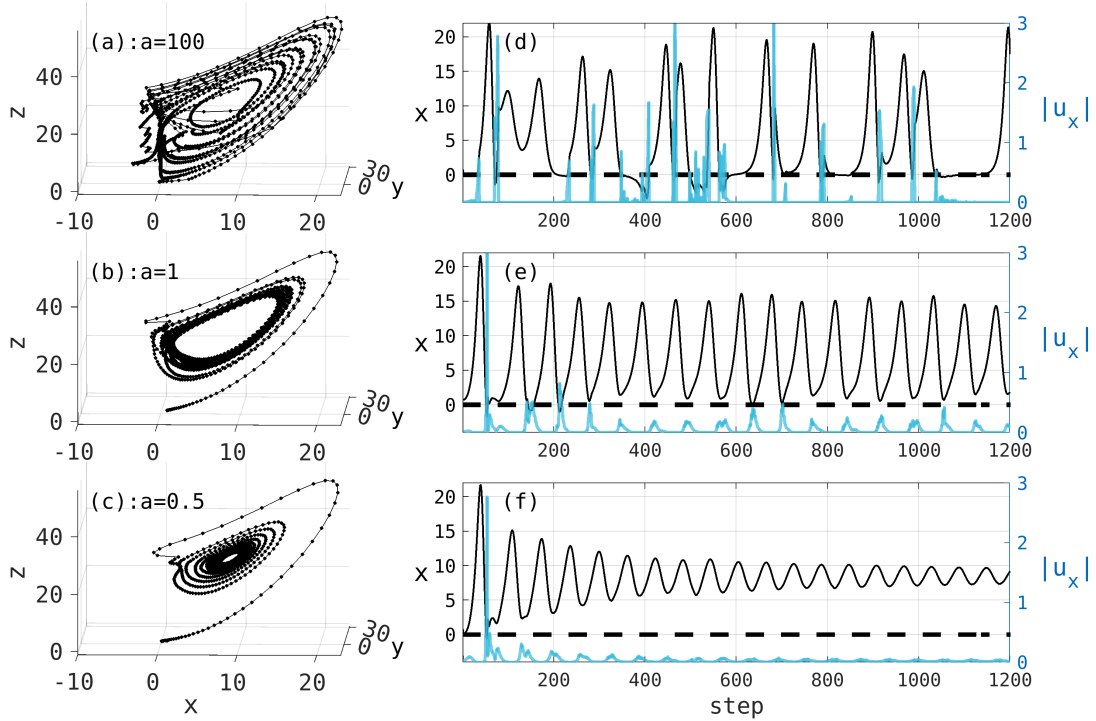


FIG. 3. Comparison of results based on different values of a in the pseudo-observation operator shown in Fig. 2. The Lorenz63 model is controlled to keep the x value positive, showing the behavior over the first 1200 steps. Panels (a), (b), and (c) show the attractors of the controlled NR for $a = 100$, $a = 1$, and $a = 0.5$, respectively. Panels (d), (e), and (f) show the evolution of x (left axis) in the controlled NR over time, with the blue lines indicating the control inputs $|u_x|$ (right axis).

all variables, x , y , z . The pseudo-observation operator, H^p , is set to the identity operator in this approach, meaning that the pseudo-observations directly correspond to the states of the reference trajectory without additional transformations.

5. Experimental results

In this section, we present the experimental results evaluating the performance of the proposed EnMPC using the Lorenz63 model. We compare two approaches, the penalty term approach and the trajectory tracking approach, for the control problem of restricting the state variable x to positive values. Furthermore, we examine how the choice of ensemble data assimilation methods forming



	Approach	Estimated control inputs	Prediction horizon T_p (hr)	Base DA method in EnMPC	Figure
(a)	penalty term ($y^p: x = 0$)	u_x	48	4DEnVar	Fig. 3
			6, 24, 48, 120		Fig. 4
				4DEnVar, EnKS, PS	Fig. 7a
(b)	trajectory tracking ($y^p: x, y, z$ from ref. traj.)	u_x, u_y, u_z	48		Figs. 5, 6
			6, 24, 48, 120		Fig. 7b

TABLE 1. Experimental setup

the basis of EnMPC (4DEnVar, EnKS, and PS) impacts its performance. In addition, we compare EnMPC with conventional MPC to assess its computational efficiency and control performance. Note that for the conventional MPC, we set the weighting matrix for the control input C^u to $0.01I$, which matches the pseudo-observation error covariance matrix C^r . We use an ensemble size of 10 for all experiments. All experiments are conducted using MATLAB on a typical laptop.

a. Control using the penalty term approach

In the penalty term approach, we restrict x to positive values by imposing penalties on regions where $x \leq 0$. Specifically, we utilize a pseudo-observation $y^p = 0$ and a control operator $H^p(x) = \log(1 + \exp(-ax))/a$ with $a = 0.5$. In this case, we apply control only through u_x using (20).

As shown in Fig. 4a, while x fluctuates between positive and negative values in the NR, all four MPC methods generally restrict x to the $x > 0$ region. This demonstrates that the proposed method successfully solves the MPC problem using ensemble approximations. In addition, the penalty term approach achieves control that takes into account constraint conditions by using the pseudo-observation operator.

The comparison of control inputs u_x shown in Fig. 4e shows that, during the initial 400 steps, the control input for EnMPC based on PS is larger than those for the other methods (4DEnVar and EnKS). As described in Sec. 2, this is because EnKS-based and 4DEnVar-based EnMPC use ensemble-based linear transformations, which help retain the statistical structure of the original ensemble (Lorenc 2003; Poterjoy and Zhang 2015; Houtekamer and Zhang 2016; Kurosawa and Poterjoy 2023). In contrast, PS-based EnMPC determines the analysis state through resampling, where particles with higher weights are replicated while those with lower weights are removed. This can lead to the analysis state being dominated by a few specific particles, potentially causing



more abrupt changes in the control input. However, this experiment uses a nonlinear observation operator $H^p(x) = \log(1 + \exp(-ax))/a$ as the penalty function, which posed challenges for EnKS-based and EnVar-based EnMPC as they inherently assume Gaussianity. In contrast, PS-based EnMPC is more appropriate for handling non-Gaussian structures and is less affected by such assumptions (Poterjoy 2016; Poterjoy et al. 2019; Kurosawa and Poterjoy 2021).

Beyond step 400, the success rate of control approaches nearly 100% for all MPC methods, and during this period, the magnitudes of control inputs for the three EnMPC methods show no significant differences. This suggests that the choice of data assimilation method influences the performance especially during the initial stages.

When comparing conventional MPC and EnMPC, it becomes clear that EnMPC achieves significantly reduced control input magnitudes, which leads to smaller oscillations compared to conventional MPC. This is likely because conventional MPC uses a fixed control weight matrix \mathbf{C}^u in (1), whereas EnMPC estimates it from the analysis ensemble as \mathbf{P}^a in (16).

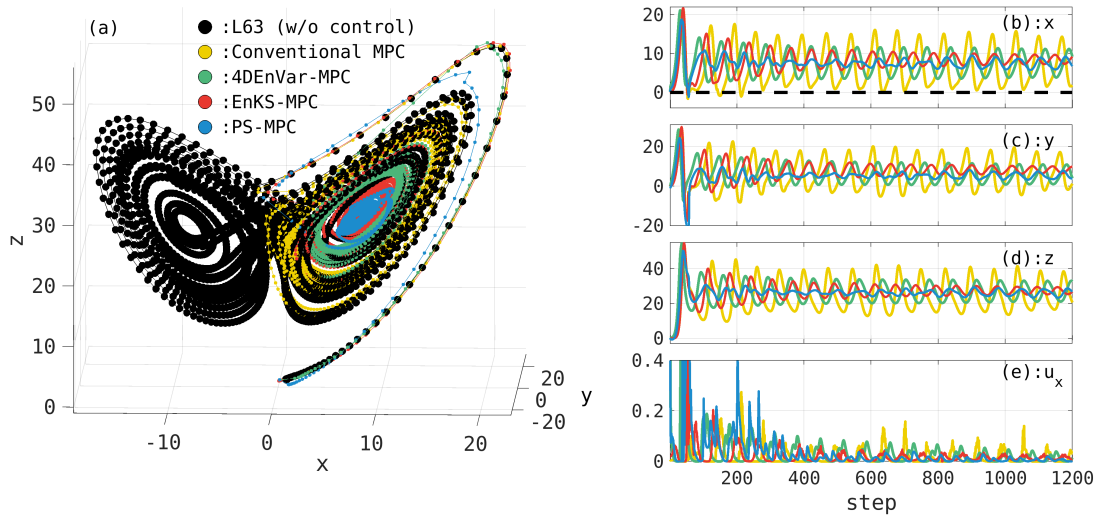


FIG. 4. Comparison of results using the conventional MPC and EnMPC with the penalty term approach. (a) The trajectory of the uncontrolled and controlled NR, (b) time series of the values of x , (c) y , and (d) z in the controlled NR, and (e) the estimated control input u_x . The black dots represent the trajectory of the uncontrolled NR, and the yellow dots show controlled NR by the conventional MPC. Green, red, and blue represent the trajectories of the NR controlled by EnMPC based on 4DEnVar, EnKS, and PS, respectively. The dashed line in (b) indicates the control objective, where $x > 0$.



437 *b. Control using the trajectory tracking approach*

438 The trajectory tracking approach controls the system state towards a predefined reference trajec-
439 tory that satisfies $x > 0$. We employ the trajectory data from Kawasaki and Kotsuki (2024) as the
440 reference and consider all three control variables u_x , u_y , and u_z using (21).

441 The results demonstrate that the proposed EnMPC can accurately follow the reference trajectory.
442 In particular, 4DVar-based and EnKS-based EnMPC provide smooth and stable control inputs,
443 while PS-based EnMPC requires larger control inputs (Figs. 5e–g). As mentioned in Sec. 5a,
444 this is because PS-based EnMPC uses particles to represent the distribution, whereas the other two
445 methods use ensemble-based transformations. In terms of tracking performance, PS-based EnMPC
446 achieves significantly lower root mean squared error (RMSE) of 0.22 compared to 3.04 and 3.03 for
447 4DVar-based and EnKS-based EnMPC, respectively (Fig. 6). This suggests that the PS-based
448 EnMPC, known for its flexibility in handling nonlinear regimes, can more accurately represent
449 complex behaviors like the reference trajectory. In contrast, EnKS-based and EnVar-based EnMPC
450 struggle to properly incorporate the nonlinearities of the reference trajectory, resulting in larger
451 RMSE values.

452 When compared to conventional MPC, all EnMPC methods exhibit significant advantages in
453 both tracking performance and control efficiency. Conventional MPC shows an RMSE of 5.91
454 (Fig. 6), which is considerably higher than any of the EnMPC methods, demonstrating its difficulty
455 in accurately following the reference trajectory. As discussed in Sec.5a, this is likely due to the
456 fixed control weight matrix \mathbf{C}^u in conventional MPC, which limits its flexibility in adapting to the
457 reference trajectory in the prediction horizon.

458 To enhance the accuracy of the control in both conventional MPC and EnMPC, or to reduce
459 the abrupt control inputs in PS-based EnMPC, improving the prediction horizon or increasing
460 ensemble sizes would be effective. These improvements remain an important subject for future
461 research.

468 *c. Impact of prediction horizon on computational time and control performance*

469 This section provides a comparison of the computational time required by conventional MPC and
470 various EnMPC methods across different prediction horizons (T_p). We perform the comparison
471 for both the penalty term approach (Fig. 7a) and the trajectory tracking approach (Fig. 7b).

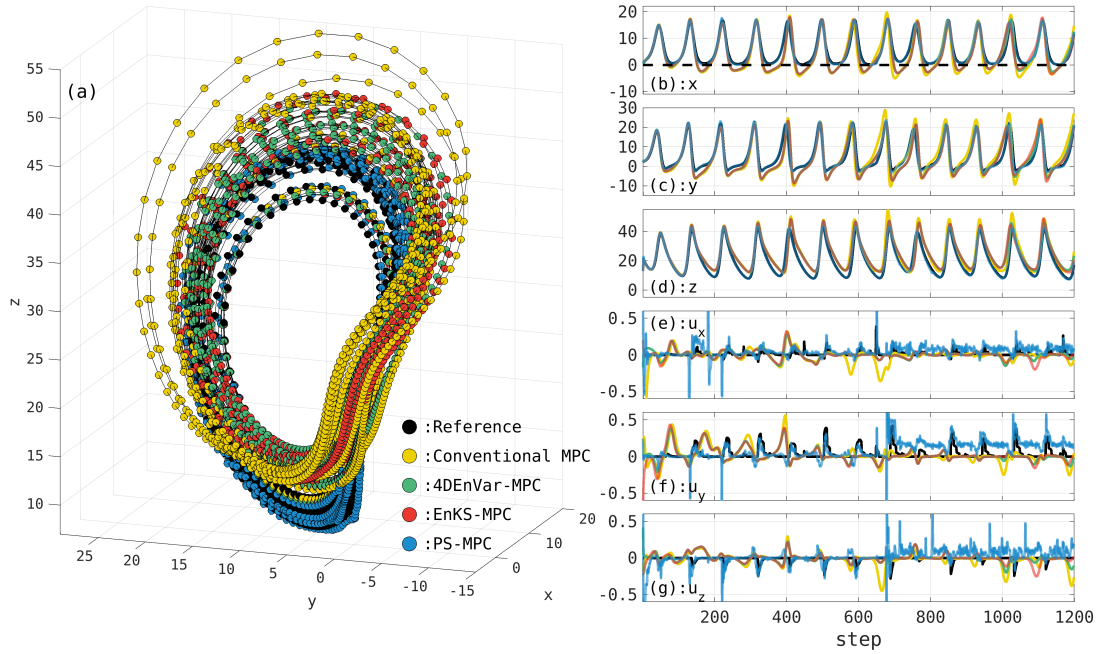


FIG. 5. As in Fig. 4, but the optimal control input values are determined to follow a reference trajectory that satisfies the constraints. The black dots represent the reference trajectory.

In the penalty term approach (Fig. 7a), EnMPC methods consistently achieve high success rates (approximately 1.0) across all prediction horizons. In contrast, conventional MPC fails to control effectively when the prediction horizon is short (6 and 24 hours). In terms of computational time, conventional MPC exhibits a sharp increase as T_p extends, reflecting its computational inefficiency due to the need for full-model evaluations to calculate optimal control inputs. For example, at $T_p = 120$ hr, the computational time for conventional MPC is 620 s. On the other hand, the EnMPC methods all show a much lower computational times, with the PS-based approach yielding 121 s, the 4DEnVar-based approach 81 s, and the EnKF-based approach being the most computationally efficient at 16 s. This is because the 4DEnVar and PS methods used in the current study require iterations to determine the optimal control inputs, whereas EnKS does not. Exploring alternative data assimilation methods to further reduce computational time remains an important future research topic.

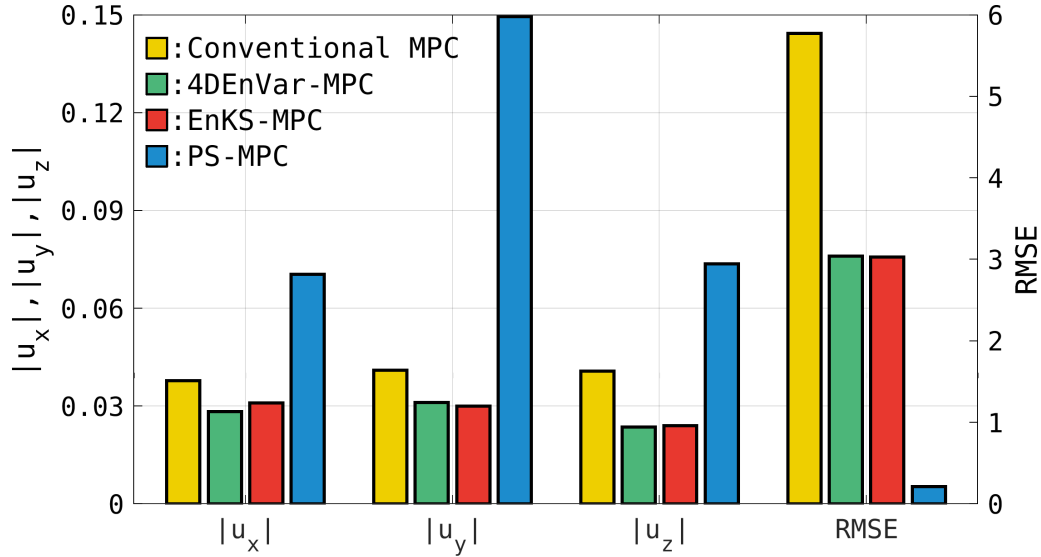


FIG. 6. Comparison of the average control input magnitudes ($|u_x|$, $|u_y|$, and $|u_z|$; left axis) and RMSE (right axis) with respect to the reference trajectory, calculated as averages from step 400 to step 2000. Yellow, green, red, and blue bars represent conventional MPC, 4DEnVar-based EnMPC, EnKS-based EnMPC, and PS-based EnMPC, respectively. These values correspond to the results in Fig. 5.

For the trajectory tracking approach (Fig. 7b), the PS-based EnMPC achieves the lowest RMSE, maintaining high control accuracy across all prediction horizons. This is because PS does not assume Gaussianity and effectively handles the nonlinear regime, making it well-suited for accurately representing complex reference trajectories. In contrast, conventional MPC exhibits significantly higher RMSE values, indicating difficulty in tracking the reference trajectory, regardless of T_p . In terms of computational time, PS-based EnMPC requires slightly higher computational costs compared to other EnMPC methods, but it remains much more efficient than conventional MPC (e.g., at $T_p = 120$ hr: conventional MPC = 651 s, 4DEnVar-based = 119 s, EnKF-based = 16 s, PS-based = 158 s). This suggests that PS-based EnMPC is a strong candidate for applications where high control accuracy is prioritized. Note that the relatively higher computational cost of PS-based EnMPC in this study is due to the iterative approach used to prevent particle degeneracy (Poterjoy et al. 2019; Poterjoy 2022). Alternative PF or PS formulations may reduce computational



costs while maintaining performance (Penny and Miyoshi 2016; van Leeuwen et al. 2019; Kotsuki et al. 2022).

In summary, these results demonstrate that EnMPC outperforms conventional MPC in both computational efficiency and control performance. Particularly for longer prediction horizons, EnMPC effectively limits computational cost increases while maintaining high control accuracy.

6. Conclusion

The current study proposes EnMPC, a nonlinear control framework that combines MPC with ensemble data assimilation. EnMPC reduces computational cost while maintaining accurate control of nonlinear systems by using ensemble approximation. EnMPC assimilates pseudo-observations in a manner similar to actual observations in data assimilation to reflect constraints or reference trajectories of control problems. This unique approach provides an effective and flexible solution for addressing the challenges posed by complex and high-dimensional systems, such as those in meteorology and weather control.

We introduce two methods within the EnMPC framework: the penalty term approach and the trajectory tracking approach. The penalty term approach imposes penalties when the system violates constraints, ensuring the system remains within acceptable behavior. In contrast, the trajectory tracking approach guides the system to follow a pre-defined trajectory that is designed to satisfy the constraints. Both approaches demonstrate their effectiveness in controlling the chaotic dynamics of the Lorenz63 model, showing their potential to manage complex system behavior and their adaptability to diverse control objectives. The choice between these two approaches depends on the specific control problem. Selecting the appropriate method based on its characteristics and objectives is essential and remains a key area for future research.

Our experiments highlight the strengths of EnMPC compared to conventional MPC, particularly in terms of computational efficiency and flexibility. This advantage is primarily due to the fact that conventional MPC relies on the full model for optimization, whereas EnMPC uses ensemble approximations. Additionally, EnMPC determines the weights for control inputs using the analysis error covariance derived from ensemble data assimilation, while conventional MPC uses fixed control weights, limiting its adaptability to varying system dynamics.

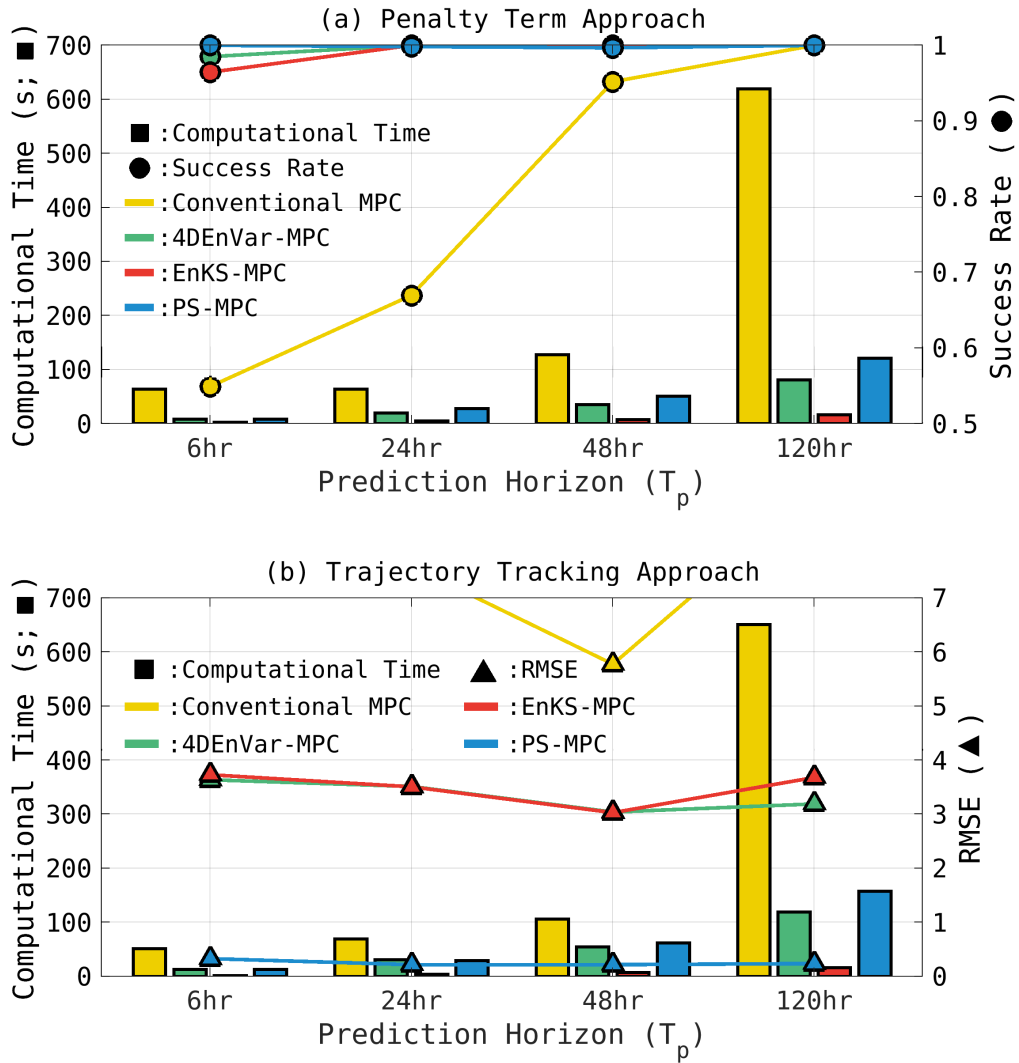


FIG. 7. Comparison of computational time and performance metrics (success rate and RMSE) as a function of the prediction horizon (T_p). Panel (a) shows the penalty term approach, depicting computational time (bars, left axis) and success rate (circles, right axis), where a higher success rate indicates more effective control. Panel (b) illustrates the trajectory tracking approach, highlighting computational time (bars, left axis) and RMSE (triangles, right axis), where a lower RMSE indicates more accurate tracking of the reference trajectory. Yellow, green, red, and blue bars represent conventional MPC, 4DEnVar-based EnMPC, EnKS-based EnMPC, and PS-based EnMPC, respectively. The values for $T_p = 48$ hours in panel (a) and (b) correspond to the results presented in Fig. 4 and 5, respectively.



532 A key aspect of our investigation involves exploring the performance of different ensemble data
533 assimilation methods that form the foundation of the EnMPC framework, which highlights the
534 importance of selecting the appropriate ensemble smoother method, such as 4DVar, EnKS, and
535 PS. For instance, while 4DVar-based and EnKS-based EnMPC provide smooth and efficient
536 control, the flexibility of PS-based EnMPC in handling nonlinear and non-Gaussian dynamics
537 leads to greater accuracy, particularly when tracking nonlinear reference trajectories.

538 Despite its advantages, EnMPC is sensitive to factors such as the pseudo-observations, prediction
539 horizon, ensemble size, and the choice of data assimilation method. For instance, achieving
540 optimal performance with the penalty term approach requires careful tuning of pseudo-observation
541 operators. The sensitivities highlight the need for further investigation and optimization to enhance
542 the effectiveness and applicability of EnMPC.

543 In conclusion, EnMPC represents a promising framework for controlling chaotic and nonlinear
544 systems, with potential applications extending to operational weather control. Future work will
545 focus on addressing the remaining challenges, including improving computational efficiency, op-
546 timizing parameter selection, and mitigating sampling errors. By advancing these areas, EnMPC
547 could become a powerful tool for operational applications, offering new possibilities for weather
548 control and beyond.

549 *Acknowledgments.* This work was supported by the Japan Science and Technology Agency Moon-
550 shot R&D program Grant Numbers #JPMJMS2389-4-1, JPMJMS2389-4-2, and JSPS KAKENHI
551 Grant Number #JP24K22969.

552 *Code/Data availability.* All software, documentation, and methods used to support this study are
553 available from the corresponding author at Chiba University.

554 *Author contribution.* KK, AO, FK, and SK conceptualized this study. KK conducted the nu-
555 merical experiments and wrote the paper. AO, FK, and SK provided comments that improved the
556 clarity of the manuscript.

557 *Competing interests.* The contact author has declared that neither of the authors has any competing
558 interests.



References

- Babu, M., R. R. Theerthala, A. K. Singh, B. Baladhurgesh, B. Gopalakrishnan, K. M. Krishna, and S. Medasani, 2019: Model predictive control for autonomous driving considering actuator dynamics. *2019 American Control Conference (ACC)*, 1983–1989, <https://doi.org/10.23919/ACC.2019.8814940>.
- Bannister, R. N., 2017: A review of operational methods of variational and ensemble-variational data assimilation. *Q.J.R. Meteorol. Soc.*, **143** (703), 607–633, <https://doi.org/10.1002/qj.2982>.
- Bishop, C. H., B. J. Etherton, and S. J. Majumdar, 2001: Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects. *Mon. Wea. Rev.*, **129** (3), 420 – 436, [https://doi.org/10.1175/1520-0493\(2001\)129<0420:ASWTET>2.0.CO;2](https://doi.org/10.1175/1520-0493(2001)129<0420:ASWTET>2.0.CO;2).
- Buehner, M., 2005: Ensemble-derived stationary and flow-dependent background-error covariances: Evaluation in a quasi-operational NWP setting. *Q.J.R. Meteorol. Soc.*, **131** (607), 1013–1043, <https://doi.org/10.1256/qj.04.15>.
- Chopin, N., and O. Papaspiliopoulos, 2020: *Particle Smoothing*, 189–227. Springer International Publishing, Cham, https://doi.org/10.1007/978-3-030-47845-2_12.
- Courtier, P., J.-N. Thépaut, and A. Hollingsworth, 1994: A strategy for operational implementation of 4D-Var, using an incremental approach. *Q.J.R. Meteorol. Soc.*, **120** (519), 1367–1387, <https://doi.org/10.1002/qj.49712051912>.
- Evensen, G., 1994: Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *J. Geophys. Res.: Oceans*, **99** (C5), 10 143–10 162, <https://doi.org/10.1029/94JC00572>.
- Evensen, G., 2009: *Data assimilation: the ensemble Kalman filter*. Springer Science & Business Media.
- Fairbairn, D., S. R. Pring, A. C. Lorenc, and I. Roulstone, 2014: A comparison of 4DVar with ensemble data assimilation methods. *Q.J.R. Meteorol. Soc.*, **140** (678), 281–294, <https://doi.org/10.1002/qj.2135>.



- 585 Houtekamer, P. L., and H. L. Mitchell, 1998: Data assimilation using an ensemble Kalman
 586 filter technique. *Mon. Wea. Rev.*, **126** (3), 796 – 811, [https://doi.org/10.1175/1520-0493\(1998\)](https://doi.org/10.1175/1520-0493(1998)126<0796:DAUAEK>2.0.CO;2)
 587 126<0796:DAUAEK>2.0.CO;2.
- 588 Houtekamer, P. L., and F. Zhang, 2016: Review of the ensemble Kalman filter for atmospheric data
 589 assimilation. *Mon. Wea. Rev.*, **144** (12), 4489 – 4532, [https://doi.org/10.1175/MWR-D-15-0440.](https://doi.org/10.1175/MWR-D-15-0440.1)
 590 1.
- 591 Hunt, B. R., E. J. Kostelich, and I. Szunyogh, 2007: Efficient data assimilation for spatiotemporal
 592 chaos: A local ensemble transform Kalman filter. *Physica D: Nonlinear Phenomena*, **230** (1),
 593 112–126, <https://doi.org/10.1016/j.physd.2006.11.008>, data Assimilation.
- 594 IPCC, 2021: *Climate Change 2021: The Physical Science Basis. Contribution of Working Group*
 595 *I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge
 596 University Press, Cambridge, United Kingdom and New York, NY, USA, [https://doi.org/10.](https://doi.org/10.1017/9781009157896)
 597 1017/9781009157896.
- 598 Kaiser, E., J. N. Kutz, and S. L. Brunton, 2018: Sparse identification of nonlinear dynamics
 599 for model predictive control in the low-data limit. *Proc. R. Soc. A.*, **474** (2219), 20180335,
 600 <https://doi.org/10.1098/rspa.2018.0335>.
- 601 Kalnay, E., 2003: *Atmospheric modeling, data assimilation and predictability*. Cambridge univer-
 602 sity press.
- 603 Kawasaki, F., and S. Kotsuki, 2024: Leading the Lorenz 63 system toward the prescribed regime
 604 by model predictive control coupled with data assimilation. *Nonlin. Processes Geophys.*, **31** (3),
 605 319–333, <https://doi.org/10.5194/npg-31-319-2024>.
- 606 Kotsuki, S., T. Miyoshi, K. Kondo, and R. Potthast, 2022: A local particle filter and its gaussian
 607 mixture extension implemented with minor modifications to the LETKF. *Geosci. Model Dev.*,
 608 **15** (22), 8325–8348, <https://doi.org/10.5194/gmd-15-8325-2022>.
- 609 Kurosawa, K., S. Kotsuki, and T. Miyoshi, 2023: Comparative study of strongly and weakly
 610 coupled data assimilation with a global land–atmosphere coupled model. *Nonlin. Processes*
 611 *Geophys.*, **30** (4), 457–479, <https://doi.org/10.5194/npg-30-457-2023>.



- 612 Kurosawa, K., and J. Poterjoy, 2021: Data assimilation challenges posed by nonlinear operators:
 613 A comparative study of ensemble and variational filters and smoothers. *Mon. Wea. Rev.*, **149** (7),
 614 2369 – 2389, <https://doi.org/10.1175/MWR-D-20-0368.1>.
- 615 Kurosawa, K., and J. Poterjoy, 2023: A statistical hypothesis testing strategy for adaptively blending
 616 particle filters and ensemble Kalman filters for data assimilation. *Mon. Wea. Rev.*, **151** (1), 105
 617 – 125, <https://doi.org/10.1175/MWR-D-22-0108.1>.
- 618 Langmuir, I., 1948: The production of rain by a chain reaction in cumulus clouds at temperatures
 619 above freezing. *Journal of Atmospheric Sciences*, **5** (5), 175 – 192, [https://doi.org/10.1175/](https://doi.org/10.1175/1520-0469(1948)005<0175:TPORBA>2.0.CO;2)
 620 1520-0469(1948)005<0175:TPORBA>2.0.CO;2.
- 621 Lea, D. J., I. Mirouze, M. J. Martin, R. R. King, A. Hines, D. Walters, and M. Thurlow, 2015:
 622 Assessing a new coupled data assimilation system based on the Met Office coupled atmo-
 623 sphere–land–ocean–sea ice model. *Mon. Wea. Rev.*, **143** (11), 4678 – 4694, [https://doi.org/](https://doi.org/10.1175/MWR-D-15-0174.1)
 624 10.1175/MWR-D-15-0174.1.
- 625 Leith, C. E., 1974: Theoretical skill of Monte Carlo forecasts. *Mon. Wea. Rev.*, **102** (6), 409 – 418,
 626 [https://doi.org/10.1175/1520-0493\(1974\)102<0409:TSOMCF>2.0.CO;2](https://doi.org/10.1175/1520-0493(1974)102<0409:TSOMCF>2.0.CO;2).
- 627 Leutbecher, M., and T. Palmer, 2008: Ensemble forecasting. *J. Comput. Phys.*, **227** (7), 3515–3539,
 628 <https://doi.org/10.1016/j.jcp.2007.02.014>, predicting weather, climate and extreme events.
- 629 Liu, C., Q. Xiao, and B. Wang, 2009: An ensemble-based four-dimensional variational data
 630 assimilation scheme. Part II: Observing system simulation experiments with advanced research
 631 wrf (ARW). *Mon. Wea. Rev.*, **137** (5), 1687 – 1704, <https://doi.org/10.1175/2008MWR2699.1>.
- 632 Lorenc, A. C., 2003: The potential of the ensemble Kalman filter for NWP—a comparison with
 633 4d-var. *Q.J.R. Meteorol. Soc.*, **129** (595), 3183–3203, <https://doi.org/10.1256/qj.02.132>.
- 634 Lorenz, E. N., 1963: Deterministic nonperiodic flow. *J. Atmos. Sci.*, **20** (2), 130–141,
 635 [https://doi.org/10.1175/1520-0469\(1963\)020<0130:DNF>2.0.CO;2](https://doi.org/10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2).
- 636 Miyoshi, T., and K. Aranami, 2006: Applying a four-dimensional local ensemble transform kalman
 637 filter (4d-letkf) to the jma nonhydrostatic model (nhm). *SOLA*, **2**, 128–131, [https://doi.org/](https://doi.org/10.2151/sola.2006-033)
 638 10.2151/sola.2006-033.



- 639 Miyoshi, T., and Q. Sun, 2022: Control simulation experiment with Lorenz’s butterfly attractor.
 640 *Nonlin. Processes Geophys.*, **29** (1), 133–139, <https://doi.org/10.5194/npg-29-133-2022>.
- 641 Morari, M., and J. H. Lee, 1999: Model predictive control: past, present and future. *Comput.*
 642 *Chem. Eng.*, **23** (4), 667–682, [https://doi.org/10.1016/S0098-1354\(98\)00301-9](https://doi.org/10.1016/S0098-1354(98)00301-9).
- 643 Nyobe, S., F. Campillo, S. Moto, and V. Rossi, 2023: The one step fixed-lag particle smoother as
 644 a strategy to improve the prediction step of particle filtering. *Revue Africaine de Recherche en*
 645 *Informatique et Mathématiques Appliquées*, **Volume 39-2023**, <https://doi.org/10.46298/arima>.
 646 10784.
- 647 Ouyang, M., K. Tokuda, and S. Kotsuki, 2023: Reducing manipulations in a control simulation
 648 experiment based on instability vectors with the Lorenz-63 model. *Nonlin. Processes Geophys.*,
 649 **30** (2), 183–193, <https://doi.org/10.5194/npg-30-183-2023>.
- 650 Penny, S. G., and T. Miyoshi, 2016: A local particle filter for high-dimensional geophysical systems.
 651 *Nonlin. Processes Geophys.*, **23** (6), 391–405, <https://doi.org/10.5194/npg-23-391-2016>.
- 652 Poterjoy, J., 2016: A localized particle filter for high-dimensional nonlinear systems. *Mon. Wea.*
 653 *Rev.*, **144** (1), 59 – 76, <https://doi.org/10.1175/MWR-D-15-0163.1>.
- 654 Poterjoy, J., 2022: Regularization and tempering for a moment-matching localized particle filter.
 655 *Q.J.R. Meteorol. Soc.*, **148** (747), 2631–2651, <https://doi.org/10.1002/qj.4328>.
- 656 Poterjoy, J., L. Wicker, and M. Buehner, 2019: Progress toward the application of a local-
 657 ized particle filter for numerical weather prediction. *Mon. Wea. Rev.*, **147** (4), 1107 – 1126,
 658 <https://doi.org/10.1175/MWR-D-17-0344.1>.
- 659 Poterjoy, J., and F. Zhang, 2015: Systematic comparison of four-dimensional data assimila-
 660 tion methods with and without the tangent linear model using hybrid background error co-
 661 variance: E4DVar versus 4DEnVar. *Mon. Wea. Rev.*, **143** (5), 1601 – 1621, <https://doi.org/10.1175/MWR-D-14-00224.1>.
 662
- 663 Potthast, R., A. Walter, and A. Rhodin, 2019: A localized adaptive particle filter within an
 664 operational NWP framework. *Mon. Wea. Rev.*, **147** (1), 345 – 362, <https://doi.org/10.1175/MWR-D-18-0028.1>.
 665



- 666 Rockett, P., and E. A. Hathway, 2017: Model-predictive control for non-domestic buildings: a
 667 critical review and prospects. *Building Research & Information*, **45** (5), 556–571, [https://doi.org/](https://doi.org/10.1080/09613218.2016.1139885)
 668 10.1080/09613218.2016.1139885.
- 669 Ryan, B., and W. D. King, 1997: A critical review of the australian experience in cloud seeding.
 670 *Bull. Amer. Meteor. Soc.*, **78** (2), 239 – 254, [https://doi.org/10.1175/1520-0477\(1997\)078<0239:](https://doi.org/10.1175/1520-0477(1997)078<0239:ACROTA>2.0.CO;2)
 671 ACROTA>2.0.CO;2.
- 672 Sawada, Y., 2024a: Ensemble Kalman filter in geoscience meets model predictive control. *SOLA*,
 673 **20**, 400–407, <https://doi.org/10.2151/sola.2024-053>.
- 674 Sawada, Y., 2024b: Quest for an efficient mathematical and computational method to explore
 675 optimal extreme weather modification. arXiv, 2405.08387.
- 676 Schwenzer, M., M. Ay, T. Bergs, and D. Abel, 2021: Review on model predictive control: an
 677 engineering perspective. *Int J Adv Manuf Technol*, **117**, 1327 – 1349, [https://doi.org/10.1007/](https://doi.org/10.1007/s00170-021-07682-3)
 678 s00170-021-07682-3.
- 679 Silverman, B. A., 2001: A critical assessment of glaciogenic seeding of convective clouds for
 680 rainfall enhancement. *Bull. Amer. Meteor. Soc.*, **82** (5), 903–923.
- 681 Slingo, J., and T. Palmer, 2011: Uncertainty in weather and climate prediction. *Phil. Trans. R. Soc.*
 682 A., **369** (1956), 4751–4767, <https://doi.org/10.1098/rsta.2011.0161>.
- 683 Sluka, T. C., S. G. Penny, E. Kalnay, and T. Miyoshi, 2016: Assimilating atmospheric observations
 684 into the ocean using strongly coupled ensemble data assimilation. *Geophys. Res. Lett.*, **43** (2),
 685 752–759, <https://doi.org/10.1002/2015GL067238>.
- 686 Sun, Q., T. Miyoshi, and S. Richard, 2023: Control simulation experiments of extreme events with
 687 the Lorenz-96 model. *Nonlin. Processes Geophys.*, **30** (2), 117–128, [https://doi.org/10.5194/](https://doi.org/10.5194/npg-30-117-2023)
 688 npg-30-117-2023.
- 689 Talagrand, O., 2014: 4D-VAR: four-dimensional variational assimilation. *Advanced Data Assimilation for Geosciences: Lecture Notes of the Les Houches School of Physics: Special Issue, June 2012*, Oxford University Press, <https://doi.org/10.1093/acprof:oso/9780198723844.003.0001>.



- 692 van Leeuwen, P. J., H. R. Künsch, L. Nerger, R. Potthast, and S. Reich, 2019: Particle filters
 693 for high-dimensional geoscience applications: A review. *Q.J.R. Meteorol. Soc.*, **145** (723),
 694 2335–2365, <https://doi.org/doi.org/10.1002/qj.3551>.
- 695 Wang, B., J. Liu, S. Wang, W. Cheng, L. Juan, C. Liu, Q. Xiao, and Y.-H. Kuo, 2010: An
 696 economical approach to four-dimensional variational data assimilation. *Adv. Atmos. Sci.*, **27** (4),
 697 715–727, <https://doi.org/10.1007/s00376-009-9122-3>.
- 698 Whitaker, J. S., and T. M. Hamill, 2002: Ensemble data assimilation without perturbed observa-
 699 tions. *Mon. Wea. Rev.*, **130** (7), 1913 – 1924, [https://doi.org/10.1175/1520-0493\(2002\)130\(1913:](https://doi.org/10.1175/1520-0493(2002)130(1913:EDAWPO)2.0.CO;2)
 700 [EDAWPO\)2.0.CO;2](https://doi.org/10.1175/1520-0493(2002)130(1913:EDAWPO)2.0.CO;2).
- 701 Zhang, F., M. Zhang, and J. A. Hansen, 2009: Coupling ensemble Kalman filter with four-
 702 dimensional variational data assimilation. *Adv. Atmos. Sci.*, **26** (1), 1–8, [https://doi.org/10.1007/](https://doi.org/10.1007/s00376-009-0001-8)
 703 [s00376-009-0001-8](https://doi.org/10.1007/s00376-009-0001-8).
- 704 Zhu, S., and Coauthors, 2022: A four-dimensional ensemble-variational (4DEnVar) data assim-
 705 ilation system based on GRAPES-GFS: System description and primary tests. *J. Adv. Model.*
 706 *Earth Syst.*, **14** (7), e2021MS002 737, <https://doi.org/10.1029/2021MS002737>.
- 707 Zupanski, M., 1996: A preconditioning algorithm for four-dimensional variational data assim-
 708 ilation. *Mon. Wea. Rev.*, **124** (11), 2562 – 2573, [https://doi.org/10.1175/1520-0493\(1996\)](https://doi.org/10.1175/1520-0493(1996)124(2562:APAFD)2.0.CO;2)
 709 [124\(2562:APAFD\)2.0.CO;2](https://doi.org/10.1175/1520-0493(1996)124(2562:APAFD)2.0.CO;2).