# Resilient estimation of state and time-varying parameters by particle filter for nonlinear high-dimensional systems

Author: 佐藤　峰斗

**(Mineto Satoh)**

Supervisor: Prof. Shin'ya Nakano

**Dissertation**

submitted to the Department of Statistical Science

School of Multidisciplinary Sciences

***Doctor of Philosophy***

The Graduate University for Advanced Studies, SOKENDAI

March 2025

# Acknowledgments

I would like to express my sincere gratitude to Professor Shin'ya Nakano. The completion of this research would not have been possible without the professor's expertise.

I would also like to express my sincere thanks to Professor Peter Jan Van Leeuwen. Most of the work in Chapter 3 is based on my collaborative research with the professor while I was a visiting scholar in residence at the Data Assimilation Research Center at the University of Reading. I would also like to thank the people at the institution who supported me during my stay with discussions and computing environments.

I would also like to express my gratitude to the people at The Institute of Mathematical Statistics (ISM) and The Graduate University for Advanced Studies (SOKENDAI), who provided me with guidance and research support.

I would also like to thank the NEC Corporation for agreeing to allow me to devote my time to doctoral research and for their support of my work.

Lastly, I would like to thank my parents. My father, who supported my PhD, passed away while I was still in school, but I hope this doctoral dissertation will reach him. I am also indebted to my mother, who tried her best not to cause me any inconvenience so that I could concentrate on my work and doctoral research.

# Publication

Chapter 3 of this thesis is reproduced from the following publication:

[SvLN24] Mineto satoh, Peter Jan Van Leeuwen, and Shin'ya Nakano. "Online state and time-varying parameter estimation using the implicit equal-weights particle filter." Quarterly Journal of the Royal Meteorological Society (2024).

Mineto Satoh carried out all work undertaken in this paper, and the role (contribution) is as follows: Conceptualization, investigation, methodology, software, validation, and writing the original draft. Co-authors provided supervision, base program, review, and editing support.

# Abstract

This study concerns a data assimilation process that simultaneously estimates states and parameters using collected observations. Because parameters in a numerical model are representations in the model and are unobserved quantities, parameter estimation plays an important role in obtaining accurate and reliable forecasts. Data assimilation is a procedure for incorporating observations into numerical models and obtaining the posterior distribution of the state variables, especially in high-dimensional dynamical systems. While data assimilation often focuses on generating optimal initial conditions and predicting time evolution, it is often combined with parameter estimation for model calibration. However, simultaneous estimation of states and parameters complicates the handling of nonlinearity and is challenging, especially for nonlinear high-dimensional numerical models. Furthermore, it is also important to detect changes in the characteristics of the actual system as a parameter change, but estimating time-varying parameters is even more difficult than estimating static parameters.

Therefore, this study investigated resilient estimation methods for states and time-varying parameters applicable to geophysical, climatological, and other high-dimensional applications. We focus on particle filter-based methods, considering their adaptability to nonlinearity due to the characteristics of the model and the inclusion of parameter estimation. However, particle filters have a well-known problem of the so-called degeneracy, where one particle may have a much higher weight than all the others, resulting in identical particles and a loss of diversity. Thus, we use the implicit equal-weights particle filter (IEWPF). This method eliminates the need for resampling in general particle filters by equalizing all particles, thus avoiding the degeneracy problem. In this study, we first combined the parameter vector with the state vector using an augmented state space model. Then, for resilient estimation of time-varying pa-

rameters, a machine learning-inspired nudging algorithm was incorporated into the time-evolution model for the parameters. This algorithm "nudges" model parameters toward their plausible values. Next, we established an adaptive determination method for the coefficient specific to IEWPF. This method makes it unnecessary to give the pre-set values and is resilient to nonlinearity and parameter errors. Finally, the proposed method was validated using a linear model and a nonlinear Lorenz 96 model. The proposed method achieved simultaneous, degeneracy-free estimation for 1000-dimensional state variables and 3-dimensional time-varying parameters with only 20 particles.

# Contents

# 1 Introduction

## 1.1 Background and motivation

### 1.1.1 Data assimilation

In recent years, extreme weather events, such as unusually high temperatures, heavy rainfall, and crop failures, which are believed to be caused by climate change, have become a global threat. Climate change is a long-term shift in weather patterns from the tropics to the poles, which stresses not only habitats and agriculture but also various other sectors.

To scientifically understand the effects of climate change and to predict its future impacts, it is useful to use numerical models for the processes that govern the climate system. Many aspects of the Earth's climate system are chaotic, and their time evolution is sensitive to small perturbations to the initial conditions. In addition, modeling errors (i.e., model uncertainties) must be taken into account due to the limitations of representing the climate system in numerical models. Therefore, medium- to long-

term predictions of climate are challenging. To make reliable forecasts and estimate forecast errors in the presence of both initial conditions and model uncertainty, ensemble prediction [PMM+93], a method of making several different forecasts from different initial conditions, is used.

Then, data assimilation is used to incorporate observed data into model-based ensemble prediction e.g., [Eve94]. Broadly defined, data assimilation is a statistical method that combines past knowledge about a system as a numerical model with present information about the system as observed data. From a statistical perspective, data assimilation is based on Bayes' theorem. Past knowledge can be expressed in the form of a state space model, taking into account uncertainties in the model and initial conditions. Bayes' theorem then tells us how the observations update the probability density function (pdf). Data assimilation is operationally used in numerical weather forecasting, but its other uses are much more extensive, including oceanography, hydrology, and seismology (i.e., geosciences). Specific uses of data assimilation in geosciences are summarized in [CBBE18]. One aspect of the difficulty of data assimilation in geosciences is computational difficulty due to the high dimensionality of variables and nonlinear dynamics. Therefore, because the number of ensemble members used for forecasting is often limited by computational resources, an important research question is how to improve forecasting accuracy using fewer ensemble members.

### 1.1.2 Parameter estimation in data assimilation

Although data assimilation usually focuses on generating an optimal initial state and forecasting the temporal evolution of time-varying model state variables, parameter estimation is often combined to calibrate the models (i.e., to estimate the appropriate model characteristics). Parameters here refer to model parameters expressed as tuning terms in the modeled system, which must be estimated (i.e., calibrated) to match real-world characteristics. In the following, we discuss three issues regarding parameter estimation in data assimilation.

First, because most parameters cannot be measured directly, they can only be estimated from the relationship between the parameter and the state variable. If the covariance between the observed variable and the parameter is significant, the parameter

can be estimated accurately because it strongly influences the observed variable. Conversely, if the observed variable is not strongly correlated with the parameter value, the parameter cannot be estimated well. Thus, it can be seen that parameter estimation depends on the relationship between the variables and parameters as determined by the simulation model and the error covariance set by the data assimilation process.

Second, as with variables, the parameters to be estimated may also be high-dimensional. According to [RPM13], the typical number of parameters that can be adjusted in a geophysical numerical model is at least $O(10^2)$ even without considering spatial variability. Thus, because the cost of exploring the entire parameter space is prohibitive, in many cases, most parameters are fixed to preset values, and only a few are subjectively adjusted manually. In other words, it is assumed to estimate parameters with sufficiently low dimensions compared to the dimensions of the variables.

Finally, it is important to estimate changes in the characteristics of the real system as changes in parameters, that is, parameters are considered not only static but also time-varying. An example of the purpose of introducing time-varying parameters is to find when the operating conditions for the system change or when some failure occurs, as a change in the model parameters. To detect system changes, it is important to be able to detect abrupt changes in certain parameter values. According to [ZMC+17], state and parameter estimation plays an important role in the application of process monitoring, online optimization, and process control. The difficulty of these applications is identifying changes in model parameters when the operating conditions for the processing system have changed, or some faults have occurred in the processing system. As another example, in hydrological modeling, parameters are usually assumed to be constant and calibrated using a particular data record to obtain an optimal parameter set or stationary parameter distributions. However, it has been reported that calibration (i.e., data assimilation) over a specific time period, assuming time-varying parameters, improves accuracy [DLG+16]. The above examples show that estimating time-varying parameters improves forecast accuracy and plays an important role in determining when model characteristics change abruptly. However, when parameters may change abruptly, it is challenging to estimate them as rapidly and stably as possible after the observation. This is due to the potential trade-off between the ability to follow abrupt changes and achieve robustness. From the above examples, simultaneous estimation of state and time-varying parameters is expected to play an important

role in applications. However, even when the model itself has linear dynamics, the problem is complex because the introduction of parameters as additional variables can make the problem nonlinear [EDS98].

### 1.1.3 Motivation

Given the above background, this study investigated a method for simultaneously estimating variables and parameters that can be applied to geophysical, climatological, and other high-dimensional cases, focusing on nonlinear time-varying systems with large state-vector dimensions. Among the many data assimilation methods, we focus on particle filter-based methods, considering their adaptability to nonlinearity due to the characteristics of the model. Below, the advantages of using particle filters for parameter estimation in data assimilation are presented. Because the particle filter considers a general pdf without the Gaussian assumption, it can overcome the limitations of the four-dimensional variational method (4DVar) and the ensemble Kalman filter (EnKF) when the model response to the parameters is strongly nonlinear. In fact, [VvL07] found that methods other than particle filters may produce suboptimal estimates. Also, [Kiv03] and [AT11] showed that, in experiments using a simple, highly nonlinear model, the particle filter outperformed EnKF in terms of estimating model parameters. Furthermore, particle filters are also useful for change detection and system control, as described above, because they can be applied to nonlinear, non-Gaussian state-space models [ADST04]. Therefore, particle filters are promising in that they better represent the uncertainty of parameters, especially in strongly nonlinear systems.

The effectiveness and challenges of particle filter-based methods for high-dimensional geoscience applications are summarized in [vLKN+19]. One of the key challenges in particle filters is the suppression of filter degeneracy. In the process of particle filtering, especially when the dimension of the observation is high, a particular particle may have a much higher weight than all other particles. When resampled in proportion to this weight, the replication of a particular particle becomes dominant, and the diversity of particles is lost. This is the so-called degeneracy problem. Typically, many particles (i.e., model calculation) are required to suppress the effects of degeneracy and diversify the particles. This further increases the computational cost of data assimila-

tion and parameter estimation in high-dimensional systems. Therefore, it is beneficial if degeneracy can be suppressed even with a small number of particles (i.e., with a small amount of calculation). Among the several methods proposed to prevent filter degeneracy, this study uses the implicit equal-weights particle filter (IEWPF) proposed by Zhu et al. [ZvLA16]. This method weights all the particles evenly, eliminating the need for resampling in typical particle filters and avoiding the degeneracy problem. However, it does not support parameter estimation, and there are several settings that need to be tuned (e.g., hyperparameters). Note that "parameters" in this thesis refer to the parameters of the numerical model, and other values that must be preset are referred to as hyperparameters.

## 1.2 Objectives and approach

The purpose of this study is to achieve simultaneous estimation of variables and parameters based on IEWPF and to set guidelines for the values that require prior setup or tuning. The parameters here are assumed to be initially uncertain and to vary over time, and the dimension of the parameter to be estimated is assumed to be sufficiently smaller than the dimension of the variable. The estimation of variables and parameters is based on the observation values obtained online sequentially. In other words, the aim is to improve the accuracy of the next prediction while modifying the characteristics of the system under the current observation values. Figure 1.1 shows a schematic diagram of the sequential estimation of variables and parameters in the typical data assimilation process assumed in this thesis. The top and bottom figures show the time series changes in the variables and parameters, respectively, and show how the values of the variables and parameters are updated based on the observations that are input sequentially.

In particular, the key point is whether it is possible to appropriately estimate the value of a parameter that cannot be observed and how quickly this can be done (i.e., in a few time steps). In addition, it is assumed that parameters change over time. To summarize the above, the requirements for the simultaneous estimation of variables and parameters in this study are as follows:

1. Estimation is applicable to high-dimensional and nonlinear models for geophys-

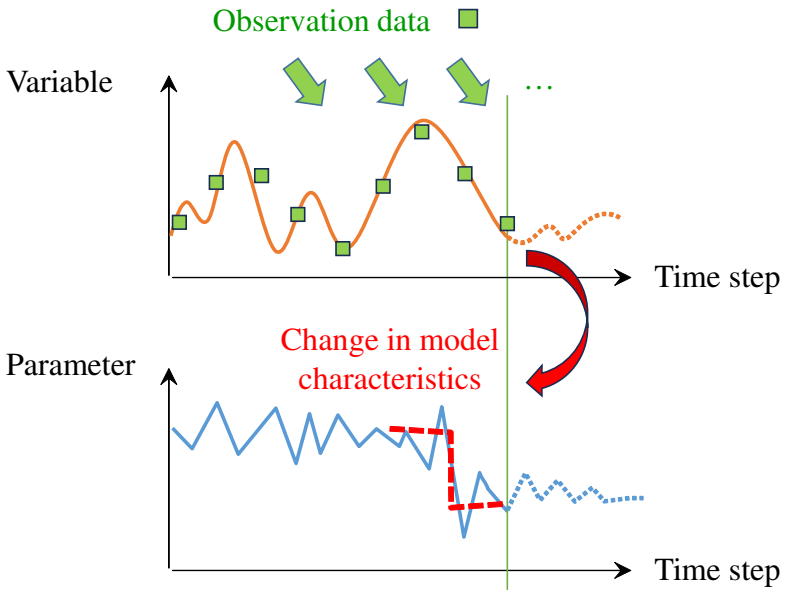


Figure 1.1: Schematic diagram of sequential estimation of variables and parameters in a typical data assimilation process. The top and bottom plots show the time series of the variable and parameters, respectively, and the green squares represent observation data. The change in model characteristics is shown for the parameters.

ical, climatological, and other high-dimensional applications.

2. The variables and parameters are estimated sequentially every time that new time series observation data are obtained.

3. Parameters that are uncertain or change over time are quickly estimated.

4. Guidelines or methods are provided for setting values that require prior setup or tuning.

To achieve the above objectives, we propose an IEWPF-based parameter estimation method. First, we combine the parameter vector with the state vector of the IEWPF using an augmented state space model for simultaneous estimation. To improve the estimation efficiency by relating variables and parameters, we introduce a covariance matrix with correlations between variables and parameters. We then incorporate an optimization algorithm from machine learning into the time evolution model of the

parameters by taking advantage of the flexibility of the proposal density in particle filtering. Specifically, this is a method of nudging parameters toward plausible values, or high likelihood.

Then, we show the guidelines for setting the inherent coefficients in IEWPF. IEWPF has a specific coefficient that determines the spread of particles, which is obtained by solving an equation to make the weight of all particles equal. However, it is not computationally appropriate to numerically solve this equation at each time step for each particle. Therefore, Zhu et al. [ZvLA16] used an analytical solution based on the so-called Lambert W function [CGH+96], but because this is a multivalued function, the solution cannot be uniquely determined. This thesis presents a method for estimating appropriate values without making prior assumptions or settings regarding this coefficient.

## 1.3 Outline

The thesis is structured as follows. Chapter 2 briefly summarizes particle filters for data assimilation. In general, data assimilation methods can be broadly divided into the following three categories: sequential, variational, and hybrid. All of these methods update the values of the state variables based on Bayes' theorem using observed data, but they are used differently depending on factors such as the linearity/nonlinearity of the problem, the number of dimensions, the cycle of observations and estimations, and the allowable computational load. We also introduce methods for estimating parameters in state space models.

Chapter 3 shows how to extend IEWPF to parameter estimation. For sequential and quick estimation, we introduce an augmented state-space model, and the parameter nudging scheme inspired by an optimization algorithm in machine learning. The validity of the method was verified under high dimensionality (1000 dimensions) and a small number of particles (20 particles) using the linear and Lorenz 96 models [Lor96].

Chapter 4 shows how to estimate the inherent coefficient in IEWPF from the perspective of eliminating the need for prior assumptions and tuning. Although the analytical solution of the Lambert W function is used, the factor can be determined adaptively by iterative calculations with low computational load. Experiments with the linear and Lorenz 96 models were performed to validate the proposed method. It is

also demonstrated that accuracy is equal to or better than that of the case with a priori assumptions (i.e., preset).

Chapter 5 reports the conclusions and discusses possible future work.

# 2

# Review of data assimilation and parameter estimation method

This chapter presents an overview of data assimilation methods based on particle filters (PFs). Then, an overview of parameter estimation methods that can be combined with PFs is presented.

## 2.1 Data assimilation methods

### 2.1.1 Introduction

Data assimilation is used to incorporate observations into prior information obtained through numerical model simulations, and it produces the best possible description of the target system and its uncertainties. The purpose of using data assimilation is often to compute the most plausible estimate of the model states or model parameters. In some cases, we would like to find the best descriptions of combinations of uncertain

state variables, parameters, or all of them together [EVvL22]. In a broader sense, or mathematically, data assimilation is a method of combining past knowledge according to Bayes' theorem [Bay63]:

$$p(x|y) = \frac{p(y|x)}{p(y)}p(x), \tag{2.1}$$

where $x$ represents the state of the system and $y$ denotes the observations. According to Bayes' theorem, the pdf of the state variable given observations $p(x|y)$ can be obtained by knowing the pdf for the state variable $p(x)$, the conditional pdf for the observations given the current state of the system $p(y|x)$, and the pdf for the observations $p(y)$.

Data assimilation methods can be broadly divided into the following three categories: variational approach [TC87], sequential approach based on an ensemble Kalman filter (EnKF) [Eve94], [HM98] and Monte Carlo methods, and hybrid approaches that combine these, such as [HS00] and [CLB13]. These methods are used differently depending on factors such as the linearity/nonlinearity of the problem, the number of dimensions, the cycle of observations and estimations, and the allowable computational load.

The sequential Monte Carlo method is also known as a PF [Kit96], and it is intended to achieve complete nonlinear data assimilation without requiring any assumptions (see e.g., [RC15], [vLKN+19]). Applying PFs to realistic high-dimensional data assimilation problems is not easy. In general, a PF requires a much larger number of particles than the variable dimension to estimate the state properly. The reason for this is that the high dimensionality of the observation vectors causes the weights to be concentrated on a single particle during resampling, the so-called degeneracy problem. In addition, nonlinear models and complex non-Gaussian distributions may require more particles to represent the probability distribution of states properly. However, the number of particles may be limited because the time integration of the simulation model to calculate the time evolution of each particle is typically computationally expensive. Therefore, an important research question is how to prevent degeneracy and improve forecasting accuracy using fewer particles.

For this reason, research, such as [vLKN+19], is being carried out on various approaches to suppress the problem of degeneracy and apply PFs to data assimilation. One example is to add a process that mitigates the degeneracy and prevents the loss of particle diversity (e.g., the merging PF [NUH07], summarized in [vL09]). Another ap-

proach is to use the so-called proposal density freedom, which controls the particles in the state space so that they obtain very similar weights, such as [DDFG+01]. Examples include an implicit PF [CMT10], equivalent-weights PF (EWPF) [vL10], and implicit equal-weights PF (IEWPF) [ZvLA16], summarized in [vLKN+19]. Another approach is localization [BSN03], [vL03], which is a standard technique in the EnKF. Localization can suppress degeneracy by limiting the impact of each observation to a localized region that is much smaller than the entire model domain.

This section focuses on methods based on PFs due to their potential application to nonlinear models and simultaneous model parameter estimation. First, the standard PF and the method using the proposal density are explained. After that, EWPF and the implicit PF are introduced, and finally, IEWPF and its revised version are explained.

In data assimilation, we consider the following state-space model in conjunction with the system and observation model:

$$\begin{aligned} x^n &= f\left(x^{n-1}\right) + \beta^n, \\ y^n &= h\left(x^n\right) + \epsilon^n, \end{aligned} \tag{2.2}$$

where $x^n$ and $y^n$ are the state variable and the observation vector at time step $n$, respectively. The function $f$ constituting the system model is a known nonlinear operator that maps the state from step $n-1$ to $n$. The function $h$ constituting the observation model is a known nonlinear observation operator. Then, $\beta^n$ and $\epsilon^n$ are random perturbations in the system and observation model, respectively.

### 2.1.2 Standard particle filter

The standard PF can be described as follows. First, consider a general state-space model, which is a generalization of Eq. (2.2), expressed as a conditional distribution as follows:

$$\begin{aligned} x^n &\sim p\left(x^n | x^{n-1}\right), \\ y^n &\sim p\left(y^n | x^n\right), \end{aligned} \tag{2.3}$$

where $x^n$ and $y^n$ are the state variable and the observation vector at time step $n$, respectively. Sequential estimation of the states represented in the state space model in Eq.

(2.3) yields the prior and filtered (posterior) distributions. Given the pdf $p(x^{n-1}|y^{1:n-1})$ for the state $x^{n-1}$, the forecast distribution for state $x^n$ at time $n$ is expressed as follows:

$$p\left(x^n|y^{1:n-1}\right) = \int p\left(x^n|x^{n-1}\right) p\left(x^{n-1}|y^{1:n-1}\right) dx^{n-1}. \tag{2.4}$$

Then, the posterior distribution is obtained from Bayes' theorem in Eq. (2.1), and we find

$$p\left(x^n|y^{1:n}\right) = \frac{p\left(y^n|x^n\right)}{p(y^n|y^{1:n-1})} p\left(x^n|y^{1:n-1}\right) = \frac{p\left(y^n|x^n\right) p\left(x^n|y^{1:n-1}\right)}{\int p\left(y^n|x^n\right) p\left(x^n|y^{1:n-1}\right) dx^n}, \tag{2.5}$$
$$\text{if} \quad y^n \perp\!\!\!\perp y^{n-1}|x^n.$$

Next, a realization of model state $x_i^n \in \mathbb{R}^{N_x}$ with dimension $N_x$ is called a particle, and the prior pdf $p\left(x^n|y^{1:n-1}\right)$ is expressed with $N$ particles as follows:

$$p\left(x^n|y^{1:n-1}\right) \approx \frac{1}{N} \sum_{i=1}^{N} \delta\left(x^n - x_i^n\right), \tag{2.6}$$

where $\delta$ is a delta function. Then, by substituting this particle representation into Eq. (2.5), the posterior distribution is obtained as

$$p\left(x^n|y^{1:n}\right) \approx \sum_{i=1}^{N} w_i \delta\left(x^n - x_i^n\right), \tag{2.7}$$

in which the particle weights $w_i$ are given by

$$w_i = \frac{p\left(y^n|x_i^n\right)}{\int p\left(y^n|x^n\right) p\left(x^n|y^{1:n-1}\right) dx^n} \approx \frac{p\left(y^n|x_i^n\right)}{\sum_j p\left(y^n|x_j^n\right)}. \tag{2.8}$$

Note that, to express the pdf properly, the sum of the weights $w_i$ of each particle $i$ from $i = 1$ to $N$ is equal to one. This makes the integral of the entire state space in the particle representation of the pdf equal to one.

The typical operation of a standard PF is as follows. First, each particle is transitioned from time step $n-1$ to $n$ using the time evolution model. Then, at time step $n$ of the observation, each weight is determined based on the observation. Assimilating

an observation at time $n$ leads to a modification of the previous weights as follows:

$$w_i^n = w_i^{n-1} \frac{p\left(y^n | x_i^n\right)}{\sum_j p\left(y^n | x_j^n\right)}. \tag{2.9}$$

In the case of the sequential importance sampling, one of the PF algorithms, as the number of assimilation steps increases, the variation in weights also increases, and the weight of a certain particle can become much higher than that of all other particles. Therefore, resampling is usually performed before the next propagation to generate equally weighted particles to prevent this. Such an algorithm is called sequential importance resampling (SIR). The resampling process can simply be accomplished by duplicating the high-weight particles and discarding the low-weight particles. After resampling, some particles may have the same value, but particle diversity is restored if the model includes a stochastic component and random perturbations for each particle. Figure 2.1 schematically illustrates the operation of the SIR PF.

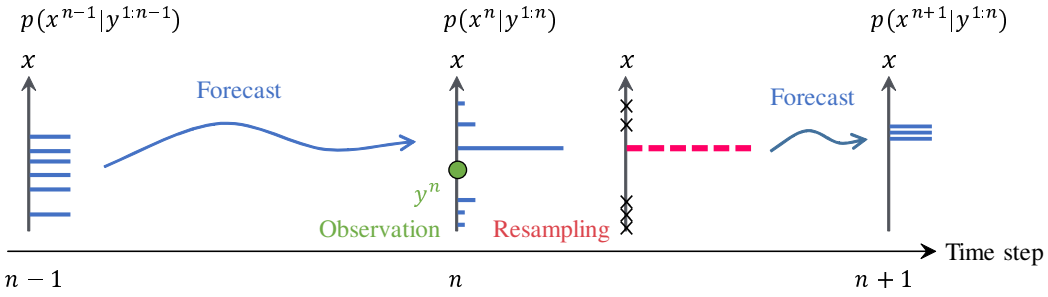


Figure 2.1: Schematic diagram of operation of standard PF. Each bar represents a particle, and the length of the bar represents the particle weight. Step $n-1$ represents equally weighted particles. Step $n$ represents particles weighted by the weight $w_i$ expressed in Eq. (2.7). The red bars represent the same number of particles replicated by resampling based on these weights, and '×' represents particles discarded. Note that step $n+1$ represents the particles of the forecast distribution and is shown partially overlapping due to degeneracy.

However, for nonlinear higher-dimensional problems, the weights can be significantly imbalanced even for a few observations, and one particle may receive a much higher weight than all the others. This problem, namely, degeneracy, is likely to become significantly pronounced as the dimension of the observation increases, and

the number of particles required has been shown to increase exponentially [SBBA08], [SBM15]. Figure 2.1 shows the case of degeneracy, which appears as an example. The weights at time step $n$ are concentrated on particles that are close to the observation, and resampling results in all particles being identical, causing loss of particle diversity at the next time step $n + 1$. From the above example, for the PF to work, the weights must remain similar and degeneracy must be suppressed.

### 2.1.3 Proposal density particle filter

Ideally, by drawing independent samples directly from the posterior pdf, all samples would have equal weight. However, this is only the case when the shape of the posterior pdf is known, and sampling from it is easy. For example, when the posterior pdf is Gaussian, and the mean and covariance can be calculated from the prior distribution using the Kalman update equations. The EnKF uses this property, so the weights of all ensemble members are equal.

The standard PF described above draws particles from the prior distribution. Then, these particles are replaced by particles representing the posterior distribution by weighting them according to their likelihood. This is a general procedure in statistics called importance sampling, which weights samples generated from a different distribution rather than the distribution of interest. The introduction mentioned that drawing from the prior data may lead to a weight that varies too much, causing the degeneracy problem. Here, the proposal density PF applies the idea of importance sampling to the transition from one time step to the next. Assuming that the numerical model is stochastic rather than deterministic gives us the freedom to modify the model equations to move the particles to those parts of the state space that we want to approach, that is, closer to the observations. Consider the forecast distribution represented as Eq. (2.4). Suppose that at time $n - 1$, there is a set of weighted particles such that

$$p\left(x^{n-1}|y^{1:n-1}\right) \approx \sum_{i=1}^{N} w_i^{n-1} \delta\left(x^{n-1} - x_i^{n-1}\right), \tag{2.10}$$

where $N$ is the ensemble size and $w_i^{n-1}$ is the weight of particle $i$. Then, the forecast distribution in Eq. (2.10) can be expressed as a weighted mixture of transition densities

as follows:

$$p\left(x^n|y^{1:n-1}\right) \approx \sum_{i=1}^{N} w_i^{n-1} p\left(x^n|x_i^{n-1}\right). \tag{2.11}$$

As mentioned above, we know that drawing once from $p\left(x^n|x_i^{n-1}\right)$ for each particle $i$ leads to filter degeneracy in high-dimensional systems with numerous independent observations $y$. Independent observation means that the off-diagonal term of the observation error covariance matrix is zero. Here, particles at time $n$ are allowed to arise according to an alternative transition density, as expressed in the following equation:

$$p\left(x^n|y^{1:n-1}\right) = \sum_{i=1}^{N} w_i^{n-1} \frac{p(x^n|x_i^{n-1})}{q(x^n|x_i^{n-1}, y^n)} q(x^n|x_i^{n-1}, y^n), \tag{2.12}$$

where $q\left(x^n|x_i^{n-1}, y^n\right)$ is the proposal density. By Bayes' formula, the posterior distribution can then be written as

$$\begin{aligned} p\left(x^n|y^{1:n}\right) &= \sum_{i=1}^{N} w_i^{n-1} \frac{p(y^n|x^n)}{p(y^n|y^{1:n-1})} \frac{p(x^n|x_i^{n-1})}{q(x^n|x_i^{n-1}, y^n)} q(x^n|x_i^{n-1}, y^n), \\ &= \sum_{i=1}^{N} w_i^{n-1} \frac{p(y^n|x_i^{n-1})}{p(y^n|y^{1:n-1})} \frac{p(x^n|x_i^{n-1}, y^n)}{q(x^n|x_i^{n-1}, y^n)} q(x^n|x_i^{n-1}, y^n), \end{aligned} \tag{2.13}$$

where the second line follows Bayes' theorem. Here, we define the following weights:

$$\hat{w}_i^{n-1} = w_i^{n-1} \frac{p(x_i^n|x_i^{n-1})}{q(x_i^n|x_i^{n-1}, y^n)}, \quad w_i^n = \hat{w}_i^{n-1} \frac{p(y^n|x_i^n)}{p(y^n|y^{1:n-1})}, \tag{2.14}$$

where the former is a proposal weight and the latter is a likelihood weight. If $x_i^n$ representing particles is drawn from the alternative model $q(x^n|x_i^{n-1}, y^n)$, Eq. (2.12) and Eq. (2.13) can also be written as follows, respectively:

$$p\left(x^n|y^{1:n-1}\right) \approx \sum_{i=1}^{N} \hat{w}_i^{n-1} \delta\left(x^n - x_i^n\right), \tag{2.15}$$

$$p\left(x^n|y^{1:n}\right) \approx \sum_{i=1}^{N} w_i^n \delta\left(x^n - x_i^n\right). \tag{2.16}$$

In other words, the prior and posterior distributions can each be expressed using the proposal weight and likelihood weight defined in Eq. (2.14). Note that the two weights

are related and have opposite effects on each other. A proposal density that brings model transitions closer to the observations has larger likelihood weights because the difference between observations and model states is small. However, because the model transitions move away from the original transitions, the weights of the proposals are smaller. However, a weak approach to observations keeps the proposal weight high but the likelihood weight low. Therefore, this implies that there is an optimal weight corresponding to the optimal position $x_i^n$ of each particle.

Based on this concept, a method of minimizing the variance of the particle weight is described below. First, the weight of particle $i$ as a function of the state $x_i^n$ can be expressed from Eq. (2.13) as follows, when an equally weighted ensemble is obtained at time step $n-1$:

$$\begin{aligned} w_i\left(x_i^n\right) &= \frac{p(y^n|x_i^n)}{Np(y^n|y^{1:n-1})}\frac{p(x_i^n|x_i^{n-1})}{q(x_i^n|x_i^{n-1},y^n)} \\ &= \frac{p(y^n|x_i^{n-1})}{Np(y^n|y^{1:n-1})}\frac{p(x_i^n|x_i^{n-1},y^n)}{q(x_i^n|x_i^{n-1},y^n)}. \end{aligned} \tag{2.17}$$

Next, consider the pair of random variables $(I, X^n)$ such that Prob $(I = i) = 1/N$ for particles and $X^n \sim q(x^n|x_i^{n-1}, y^n)$ is conditional on $I = i$. Then, the variance of the particle weight is obtained from $Var(W)$, where $W$ is defined as

$$W = w_I\left(X^n\right) = \frac{p(y^n|x_I^{n-1})}{Np(y^n|y^{1:n-1})}\frac{p(X^n|x_I^{n-1},y^n)}{q(X^n|x_I^{n-1},y^n)}. \tag{2.18}$$

From Eq. (2.18), a lower bound for $Var(W)$ that is determined by the variance of $p(y^n|x_i^{n-1})$ over $i$ can be obtained with equality if and only if

$$q(x^n|x_i^{n-1},y^n) = p(x^n|x_i^{n-1},y^n), \tag{2.19}$$

which is known as the optimal proposal density e.g., [DDFG+01]. The proof of optimality was later given by [SBM15].

In the following, we consider a state-space model represented by Eq. (2.2). If the model error is assumed to be Gaussian $\beta^n \sim \mathcal{N}(0, Q)$, the time evolution of state $x$ in Eq. (2.3) is obtained by

$$p(x^n|x^{n-1}) = \mathcal{N}\left(f(x^{n-1}), Q\right). \tag{2.20}$$

Furthermore, if we assume a linear observation operator $H$ instead of $h$ and assume the observation error to be Gaussian, $\epsilon \sim \mathcal{N}(0, R)$, the likelihood in Eq. (2.3) is as follows:

$$p(y^n|x_i^n) = \mathcal{N}\left(Hx_i^n, R\right). \tag{2.21}$$

A specific design of the proposal distribution is to add a relaxation or nudging term to the original equation to direct particles to the observations and to make their weights more similar, as presented in [vL10]. When the model transition $p(x^n|x^{n-1})$ is given by Eq. (2.20), we can consider the following transition:

$$x_i^n = f(x_i^{n-1}) + T\left\{y^n - Hf\left(x_i^{n-1}\right)\right\} + \hat{\beta}_i^n, \tag{2.22}$$

where $T$ is a relaxation matrix and $\hat{\beta}^n$ is a random perturbation. In this case, the following is assumed as the proposal distribution:

$$q(x^n|x_i^{n-1}, y^n) = \mathcal{N}\left(f(x_i^{n-1}) + T\left\{y^n - Hf\left(x_i^{n-1}\right)\right\}, \hat{Q}\right), \tag{2.23}$$

where $\hat{\beta}_i^n \sim \mathcal{N}\left(0, \hat{Q}\right)$. Here, $\hat{\beta}_i^n$ and $\hat{Q}$ are for the proposal distribution, so we used a notation that is distinct from Eq. (2.20).

In the case of the optimal proposal density defined as Eq. (2.19), the matrix $T$ becomes the Kalman-like gain $T = QH^T\left(HQH^T + R\right)^{-1}$. Therefore, the optimal proposal density is expressed as

$$p(x^n|x_i^{n-1}, y^n) = \mathcal{N}\left(\mu, \hat{Q}\right), \tag{2.24}$$

where

$$\begin{aligned} \mu &= f(x_i^{n-1}) + T\left\{y^n - Hf\left(x_i^{n-1}\right)\right\}, \\ \hat{Q} &= (I - TH)Q. \end{aligned} \tag{2.25}$$

In this case, the likelihood is expressed as follows:

$$p(y^n|x_i^{n-1}) = \mathcal{N}\left(Hf\left(x_i^{n-1}\right), HQH^T + R\right). \tag{2.26}$$

Comparing Eq. (2.26) with Eq. (2.21) for the standard PF, we can organize the error and covariance that characterize the likelihood function as shown in Table 2.1. From this

Table 2.1: Differences between standard and optimal proposal density PFs with respect to likelihood.

| Metric | Standard PF | Optimal proposal density PF |
|---|---|---|
| Distance | $y^n - Hx_i^n$ | $y^n - Hf(x_i^{n-1})$ |
| Covariance | $R$ | $HQH^T + R$ |

table, the dependence of the distance between predictions and observations on the likelihood is similar, but the difference is that the optimal proposal distribution depends on $Q$. In other words, in the optimal proposal distribution, the larger $Q$ is, the smaller the variance of each particle weight, but the weights are not equal. For comparison,

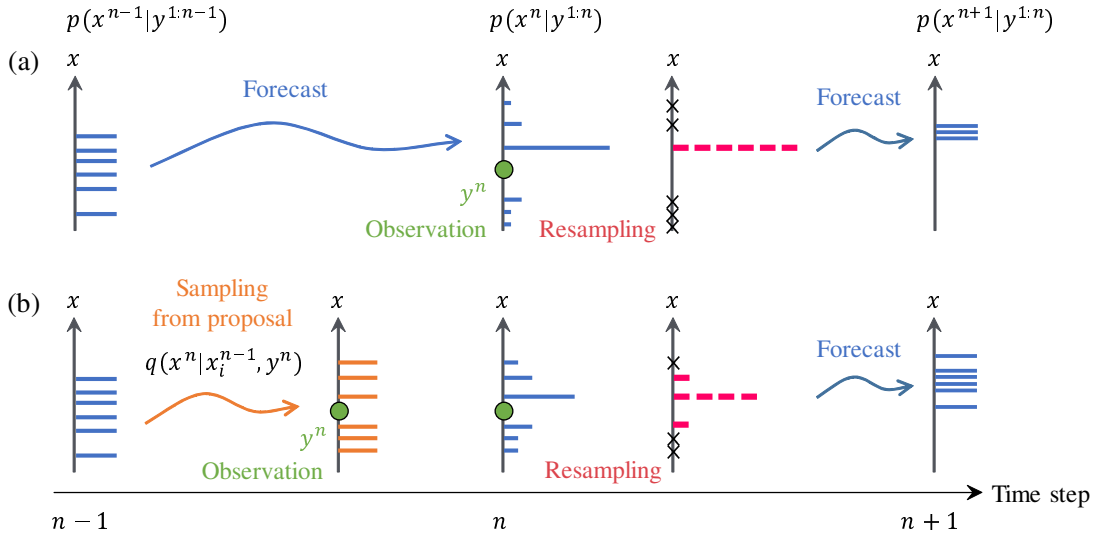


Figure 2.2: (a) Reproduction of Figure 2.1 and (b) Schematic diagram of operation of proposal density PF. Each of the orange bars represents a particle sampled from the proposal distribution $q(x^n|x_i^{n-1}, y^n)$. The blue bars in step n represent particles weighted by the weight w expressed in Eq. (2.16), and the red bars represent particles after resampling as in (a).

Figure 2.2 (a) is a reproduction of Figure 2.1 and Figure 2.2 (b) shows schematically the operation of the proposal density PF. Each particle is allowed to transition closer to the observation by the proposal distribution $q(x^n|x_i^{n-1}, y^n)$, rather than by the original model-based transition $p(x^n|x_i^{n-1})$ shown in Figure 2.2 (a). Therefore, the variation in

the weights among particles (i.e., the length of the bars in Figure 2.2) of the transition with the proposal distribution is smaller than the variation with the model transition. This indicates that the diversity of particles is relatively well maintained after resampling. Hence, the particles are less degenerate in the next forecast step and can still represent a distribution.

Particle filters using the proposal density provide flexibility for designing the proposal density. However, the optimal proposal distribution described above cannot prevent degeneracy, essentially, and the weights of the particles cannot be equal [vL10].

### 2.1.4 Implicit particle filter

The implicit PF [CMT10] is an indirect way to individually lead particles to high-probability posterior regions based on the optimal proposal density. In particular, even if the observation operator is nonlinear or proposals are used over multiple model time steps, one can draw from a Gaussian distribution and apply a transformation to that drawing to draw samples from the optimal proposal density.

The scheme draws from a Gaussian proposal $q(\xi^n) = \mathcal{N}(0, I)$, and then performs a transformation using the following equation:

$$q(\xi) = q\left(x^n | x_i^{n-1}, y^n\right) J, \tag{2.27}$$

where $\xi$ is a random perturbation and $J = \left\| \frac{\partial x^n}{\partial \xi} \right\|$ is the absolute value of the Jacobian of the transformation from $x^n$ to $\xi$. This transformation is found implicitly as follows. First, we define the following function:

$$F_i(x^n) = -\log\left[p\left(y^n | x^n\right)\ p\left(x^n | x_i^{n-1}\right)\right]. \tag{2.28}$$

Then, after drawing $\xi_i$ for each particle, we solve for $x^n$ in

$$F_i(x^n) = \frac{1}{2}\xi_i^T \xi_i + \phi_i \tag{2.29}$$

for each particle. Here,

$$\phi_i = \min_{x^n} F_i(x^n) \propto \log p\left(y^n | x_i^{n-1}\right). \tag{2.30}$$

As a result, the weight of the particle can be expressed based on $w_i^{n-1}$ in Eq. (2.13):

$$w_i^n = w_i^{n-1} \frac{\exp\left[-F_i\left(x_i^n\right)\right]}{\exp\left[-\frac{1}{2}\xi_i^T \xi_i\right]} J = w_i^{n-1} \frac{\exp\left[-F_i\left(x_i^n\right)\right]}{\exp\left[-F_i(x_i^n) + \phi_i\right]} J = w_i^{n-1} \exp\left[-\phi_i\right] J. \tag{2.31}$$

Note that the implicit mapping makes the weights dependent on the states $x_i^n$ at the current time step $n$ via the Jacobian of the transformation between $\xi$ and $x$.

However, because it is generally not easy to solve Eq. (2.29), a method using a random map was proposed by [MTAC12] as follows:

$$x_i^n = \zeta_i^n + \lambda_i(\xi_i) P^{1/2} \xi_i, \tag{2.32}$$

where $\zeta_i^n$ is $\arg\min F_i(x^n)$, $\lambda_i$ is a scalar factor, and $P$ is a covariance matrix, ideally the covariance of the posterior pdf. The key to this method is that it converts the difficult problem of directly finding $x_i^n$ into solving a scalar equation for $\lambda_i$. Hence, it is useful when the proposal distribution cannot be obtained analytically as in Eq. (2.24), that is, when the errors in both the original model and proposal density are not Gaussian or when the observation operator $H$ is nonlinear. However, this filter also cannot avoid degeneracy in principle because it uses a sampling scheme for the optimal proposal density.

### 2.1.5 Equivalent-weights particle filter

In the EWPF [vL10] [AvL13], the idea is not to sample from an exact posterior distribution, but to allow for a small error, resulting in a more equally weighted particle set. Although a single time step is described here, an extension to multiple time steps is also possible. As expressed in Eq. (2.14), because the particle weights are represented by likelihood weights and proposal weights, we can design the proposal distribution to achieve a certain weight (i.e., target weight $w_{target}$). In other words, because the particle weight is also a function of the particle position expressed as Eq. (2.17), the particle $x_i^n$ that can realize the target weight satisfies

$$w_i(x_i^n) = w_{target}. \tag{2.33}$$

Setting the target weight to a value that all particles can achieve means matching it to the particle with the lowest likelihood. So, this target weight $w$ is chosen so that a certain ratio $\rho$ of particles reaches that weight. Specifically, the weights of each particle are sorted from high to low in an array $w_i^*$, $i = \{1, 2, \ldots, N\}$ and set $w_{target} = w_{N\rho}^*$. Because there is some flexibility in setting the ratio $\rho$, we introduce how to set the target weight to the minimum weight and align all particles to this minimum weight later.

For instance, Eq. (2.33) can be solved as follows. First, the specific function of Eq. (2.33) is defined from the numerator of Eq. (2.13):

$$w_i(x_i^*) = w_i^{n-1} p\left(y^n | x_i^*\right) p\left(x_i^* | x_i^{n-1}\right) = w_{target}, \tag{2.34}$$

where $x_i^*$ is a deterministic part of the above solution, and a stochastic part must still be added. So, the final particle position is determined by adding a very small random perturbation $\xi_i^n$ from the selected position as follows:

$$x_i^n = x_i^* + \xi_i^n. \tag{2.35}$$

The addition of the above perturbation $\xi_i^n$ is based on the constraint that the support of the proposal density must contain the support of the model prior distribution. This cannot be satisfied by a deterministic part $x_i^*$ alone.

Under the conditions of Gaussian model and observation errors and a linear observation operator $H$, $x_i^*$ in Eq. (2.35) can be expressed as

$$x_i^* = f\left(x_i^{n-1}\right) + \alpha_i^* K\left\{y^n - Hf\left(x_i^{n-1}\right)\right\}, \tag{2.36}$$

where $K = QH^T(HQH^T + R)^{-1}$ is a Kalman-like gain. Substituting Eq. (2.36) into Eq. (2.34) for the constant weight $w_{target}$, we obtain a quadratic equation for $\alpha_i^*$. As its solution, $\alpha_i^*$ is expressed as

$$\alpha_i^* = 1 + \sqrt{1 - b_i/a_i}, \tag{2.37}$$

where

$$\begin{aligned} a_i &= \frac{1}{2} d_i^T R^{-1} H K d_i, \\ b_i &= \frac{1}{2} d_i^T R^{-1} d_i - \log w_i^{n-1} + \log w_{target}, \\ d_i &= y^n - Hf\left(x_i^{n-1}\right). \end{aligned} \tag{2.38}$$

Finally, the full weights of the new particles are computed, and the entire ensemble is resampled, including particles that cannot reach $w_{target}$ (but whose weights are very close to zero). Then, because the particle weights become very similar, filter degeneracy is avoided.

Let us now consider the variance of the weights. According to [vLKN+19], the variance of the weights $w$ can be estimated by the following equation:

$$Var(w) \approx \frac{1}{N^2} \frac{1-\rho}{\rho}. \tag{2.39}$$

That is, the variance of the weights depends on the number of particles and the ratio $\rho$ (i.e., the tuning parameter). For example, to keep the variance of the weights close to zero, we choose a tuning parameter $\rho$ that is close to one, that is, we let all particles reach the target weights. This represents setting of the lowest target weight, which moves particles further away from the mode of optimal proposal density and widens the posterior pdf. In contrast, as the disturbance in Eq. (2.35) is reduced, the pdf narrows. This is a limitation of this approach because we do not know what the width of the posterior pdf should be.

### 2.1.6 Implicit equal-weights particle filter

IEWPF proposed by [ZvLA16] uses implicit sampling, as in the implicit PF, to ensure that all particles have equal weight, as in EWPF. Because the weight of all particles is equal, it is possible to obtain particles that represent the posterior distribution without resampling. In IEWPF, the target weights are set equal to the minimum of the optimal proposal weights for all particles. Each particle is set to the mode of optimal proposal

density plus a scaled random perturbation, expressed as

$$x_i^n = \zeta_i^n + \alpha_i^{1/2} P^{1/2} \xi_i^n, \tag{2.40}$$

where $\zeta_i^n$ represents the mode of $q\left(x^n|x_i^{n-1}, y^n\right)$ and $P$ is a metric of the width of that pdf. The important point here is that we are not directly drawing from the proposal density $q\left(x^n|x_i^{n-1}, y^n\right)$ but from the standard Gaussian proposal density $q(\xi^n)$. Here, $q(\xi)$ is obtained by the transformation

$$q(\xi^n) = q(x^n|i, x_{1:N}^{n-1}, y^n) \left\| \frac{dx^n}{d\xi^n} \right\|, \tag{2.41}$$

where $x_{1:N}^{n-1}$ is defined as the collection of all particles at the previous step, $n-1$. This expression of the proposal distribution shows the general case where each particle has its own proposal distribution and is allowed to depend on all previous particles. This idea is similar to the implicit PF [CMT10] in that we obtain a set of particles from Eq. (2.40) instead of sampling from the posterior distribution. The key difference with regard to the IEWPF is that $\alpha_i$ is a particle-specific scale factor chosen so that the weight of each particle equals the target weight $w_{target}$. The scalar factor $\alpha_i$ is determined for each particle from

$$w_i^n(\alpha_i) = w_i^{n-1} \frac{p(x_i^n|x_i^{n-1})\, p(y^n|x_i^n)}{q(\xi_i^n)} \left\| \frac{dx_i^n}{d\xi_i^n} \right\| = w_{target}. \tag{2.42}$$

Under the assumption of a linear observation model and Gaussian error, we can choose the optimal proposal density expressed in Eq. (2.24) as $q\left(x^n|x_{1:N}^{n-1}, y^n\right)$. Then, Eq. (2.41) can be rewritten as follows:

$$q(\xi^n) = p(x^n|x_i^{n-1}, y^n) \left\| \frac{dx^n}{d\xi^n} \right\|. \tag{2.43}$$

Also, we can rewrite $p(y^n|x_i^n)\, p(x_i^n|x_i^{n-1}) = p(x_i^n|x_i^{n-1}, y^n)\, p(y^n|x_i^{n-1})$, and each distribu-

tion can be expressed as follows:

$$\begin{aligned} p(x_i^n|x_i^{n-1}, y^n) &\propto \exp\left[-\frac{1}{2}\left(x_i^n - \hat{x}_i^n\right)^T P^{-1}\left(x_i^n - \hat{x}_i^n\right)\right] \\ p(y^n|x_i^{n-1}) &\propto \exp\left[-\frac{1}{2}\phi_i\right], \end{aligned} \tag{2.44}$$

where

$$P = \left(Q^{-1} + H^T R^{-1} H\right)^{-1}, \tag{2.45}$$

$$\hat{x}_i^n = f\left(x_i^{n-1}\right) + PH^T R^{-1}\left[y^n - Hf\left(x_i^{n-1}\right)\right], \tag{2.46}$$

$$\phi_i = \left[y^n - Hf\left(x_i^{n-1}\right)\right]^T \left(HQH^T + R\right)^{-1}\left[y^n - Hf\left(x_i^{n-1}\right)\right]. \tag{2.47}$$

Then, taking the logarithm of Eq. (2.42) and using the above assumptions, we find

$$-2\log w_i^n = -2\log w_i^{n-1} + \left[-2\log\left(\frac{p(x_i^n|x_i^{n-1}, y^n)\; p(y^n|x_i^{n-1})}{q(\xi_i^n)}\left\|\frac{dx_i^n}{d\xi_i^n}\right\|\right)\right]. \tag{2.48}$$

By substituting Eq. (2.44), we obtain

$$-2\log w_i^n = -2\log w_i^{n-1} + \left(x_i^n - \hat{x}_i^n\right)^T P^{-1}\left(x_i^n - \hat{x}_i^n\right) + \phi_i - \xi_i^{nT}\xi_i^n - 2\log\left(\left\|\frac{dx_i^n}{d\xi_i^n}\right\|\right). \tag{2.49}$$

Here, if we assume that $\zeta_i^n$ in Eq. (2.40) is the mode of $p(x^n|x_i^{n-1}, y^n)$, given by $\zeta_i^n = \hat{x}_i^n$, then $x_i^n = \hat{x}_i^n + \alpha_i^{1/2} P^{1/2}\xi_i^n$ . Thus, Eq. (2.49) can be written as

$$-2\log w_i^n = -2\log w_i^{n-1} + \alpha_i \xi_i^{nT}\xi_i^n + \phi_i - \xi_i^{nT}\xi_i^n - 2\log\left(\alpha_i^{N_x/2}\left\|P^{1/2}\right\|\left|I + \frac{\xi_i^n}{\alpha_i^{1/2}}\frac{\partial\alpha_i^{1/2}}{\partial\xi_i^{nT}}\right|\right). \tag{2.50}$$

Setting the weight of all particles to the desired weight is equivalent to making all values of $\log w_i$ constant. Therefore, if we write the constant term as $C$, we can rewrite Eq. (2.50) as follows:

$$(\alpha_i - 1)\,\xi_i^{nT}\xi_i^n - 2N_x \log\alpha_i^{1/2} - 2\log\left(\left|1 + \frac{\partial\alpha_i^{1/2}}{\partial\xi_i^{nT}}\frac{\xi_i^n}{\alpha_i^{1/2}}\right|\right) = C - \phi_i. \tag{2.51}$$

The right-hand side of Eq. (2.51) expresses the log-weight offsets from the target weight for each particle $i$ and can practically be obtained from $c_i \equiv C - \phi_i = \max_j[\phi_j] -$

$\phi_i$. Note that the likelihood $p(y^n|x_i^{n-1}) \propto \exp(-\phi_i/2)$ of the previous state of the $i$-th particle differs for each particle. Therefore, the log-weight offsets $c_i = C - \phi_i$ $(i = 1, \ldots, N)$ are required for all particles to reach the target weight, and $c_i \geq 0$ from the definition.

Then, we can obtain each $\alpha_i$ by solving Eq. (2.51). However, this equation is nonlinear and complex, and it is inefficient to solve it numerically for each particle. In the high-dimensional case, Eq. (2.51) can be approximated as

$$g_i^n (\alpha_i - 1) - N_x \log \alpha_i = c_i, \tag{2.52}$$

where $g_i^n = \xi_i^{nT} \xi_i^n$. Here, we note that the Lambert W function [CGH+96] gives the solution to the following equation:

$$Ax + \log(B + x) = \log C_w, \tag{2.53}$$

as

$$x = \frac{1}{A} W\left[AC_w \exp(AB)\right] - B, \tag{2.54}$$

where $W(\cdot)$ denotes the Lambert W function. Hence, by setting $\alpha_i - 1 = x$ in Eq. (2.52), the solution $\alpha_i$ can be obtained from the analytical solution expressed in Eq. (2.54) as follows:

$$\alpha_i(g_i) = -\frac{N_x}{g_i^n} W\left[-\frac{g_i^n}{N_x} \exp\left(-\frac{g_i^n}{N_x}\right) \exp\left(-\frac{c_i}{N_x}\right)\right]. \tag{2.55}$$

Note that the Lambert W function $W(\cdot)$ has two real solutions: a positive real solution given by the $W_{-1}$ branch and a negative real solution given by the $W_0$ branch. Figure 2.3 shows example of the analytical solutions $\alpha_i(g_i)$ expressed in Eq. (2.55) as a function of $g_i$ with varying the log-weight offsets $c_i$ values. The dimension $N_x$ of the state is set to 1000, and the branches $W_0$ and $W_{-1}$ are plotted in the neighborhood of $g_i = 1000$.

Figure 2.4 schematically illustrates the operation of IEWPF. Each particle is moved from its original position to a position of equal weight according to the proposal distribution $q(x^n|x_i^{n-1}, y^n)$. Because the weights of each particle are equal, the next prior distribution is generated without resampling.

Finally, we discuss the limitations of this approach. First, this method equalizes the weights of all particles by matching them to the minimum weight of the optimal proposal density. Therefore, the posterior pdf may spread, as in the case of $\rho = 1$ in the

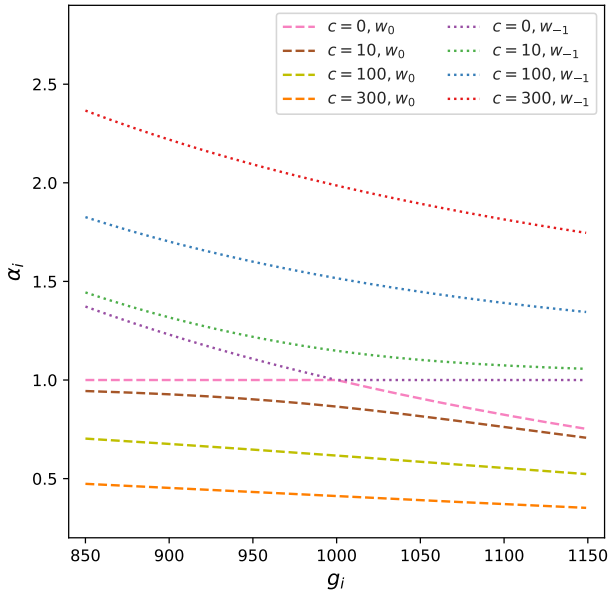


Figure 2.3: Example of the analytical solutions $\alpha_i(g_i)$ expressed in Eq. (2.55) as a function of $g_i$ with varying the log-weight offsets $c_i$: 0, 10, 100, and 300. The two branches are shown as $W_0$ (dashed line) and $W_{-1}$ (dotted line).

EWPF described above. Next, to use the analytical solution for $\alpha$, the posterior pdf is affected by the gap between its two branches (i.e., there is a forbidden area for $\alpha$). The ratio of the width of the gap $R_{gap}$ to the width of the area allowed with high probability $R_0$ is given by

$$\frac{dR_{gap}}{dR_0} \approx \frac{\sqrt{2c}}{\sqrt{N_x}}, \tag{2.56}$$

where $c_i = \max_j[\phi_j] - \phi_i$ and $N_x$ is the dimension of the state. Therefore, the relative gap width is smaller when the difference between particles in $\phi_i$ expressed in Eq. (2.47) is small and high-dimensional. If the bias caused by the above is smaller than the Monte Carlo error, then this bias does not have a direct impact.

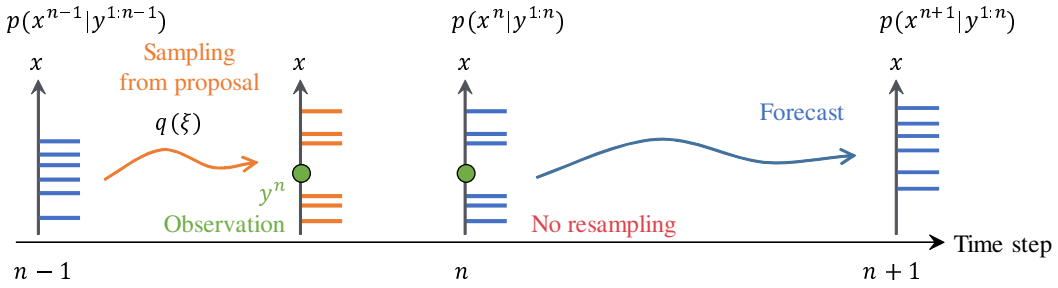


Figure 2.4: Schematic diagram of IEWPF operation. Each bar represents a particle, and the length of the bar represents the particle weight. No resampling is required because each particle is moved to equal weights rather than the original model transition.

### 2.1.7 Revised implicit equal-weights particle filter

A revised IEWPF was proposed by [SEvLA19]. In the original formulation of the IEWPF, there is a gap in the proposal distribution, as discussed above, resulting in systematic bias. The revised version uses the following two-stage proposal method, introducing a new parameter $\beta$:

$$x_i^n = \zeta_i^n + \beta^{1/2} P^{1/2} \eta_i^n + \alpha_i^{1/2} P^{1/2} \xi_i^n, \tag{2.57}$$

which is the same as Eq. (2.40) except for the $\beta^{1/2} P^{1/2} \eta_i^n$ term. Note that the scalar coefficient $\beta$ is independent of the particle, and the newly introduced perturbation vector $\eta_i$ satisfies $E\left(\xi_i^T \eta_i\right) = 0$. Then, the equal-weight equation using the transformation expressed in Eq. (2.57) replaces Eq. (2.51):

$$(\alpha_i - 1)\, \xi_i^{nT} \xi_i^n - 2N_x \log \alpha_i^{1/2} - 2 \log \left( \left| 1 + \frac{\partial \alpha_i^{1/2}}{\partial \xi_i^{nT}} \frac{\xi_i^n}{\alpha_i^{1/2}} \right| \right) = C - \phi_i - (\beta - 1) \eta_i^{nT} \eta_i^n. \tag{2.58}$$

This equation is equivalent to the original IEWPF equation, with the offset defined as

$$c_i = \max_j [D_j] - D_i, \;\; D_i = \phi_i - (1 - \beta) \eta_i^T \eta_i. \tag{2.59}$$

The additional perturbation $\eta_i$ and the common scale factor $\beta$ control the updated particle spread to be appropriate while maintaining equal weights. In practice, $\beta$ is considered a tuning parameter. Skauvold et al. [SEvLA19] provided an example of using particle ranks or coverage probabilities to determine the appropriate value of $\beta$.

## 2.2 Parameter estimation

Although data assimilation usually focuses on generating an optimal initial state and forecasting the temporal evolution of time-varying model state variables, parameter estimation is often combined to calibrate the models (i.e., to estimate the appropriate model characteristics). Several methods for estimating model parameters using data assimilation techniques exist, are effective for large, complex models (e.g., climate system models), and have positive effects on forecasts. The four-dimensional variational (4D-Var) method, the Kalman filter (KF), the EnKF, and PFs as representative examples of the algorithms are shown in [RPM13]. 4D-Var optimizes the model state within a time window to match the observations. It can account for the spatial and temporal dependence of parameters, but the high computational cost due to iterations can be an issue. The extended KF, which is an extension of KF for nonlinear models, may reduce the accuracy of estimation when nonlinearities are strong. EnKF may also be effective for nonlinear parameter estimation and can be easily parallelized, but its accuracy degrades when the number of ensemble members is insufficient. Particle filters can handle nonlinear and non-Gaussian distributions, but computational cost is an issue for high-dimensional systems. Another difficulty is that there is not necessarily only one local optimal value for a parameter, especially in time-varying nonlinear models [Mee01].

The difficulty in such parameter estimation is mainly due to the following points. First, model parameters are virtual variables that depend on modeling and parameterization, and their true values cannot be observed. Second, the influence of parameters can only be obtained through errors between model predictions and observed values, that is, it depends on the degree of influence (sensitivity) of the parameters. Finally, in terms of data assimilation, it is difficult to distinguish between errors in variables and errors in parameters. Therefore, many studies have been reported, such as [EVvL22] and [RPM13]. Particle filter-based parameter estimation methods are summarized in, for example, in [CGM07].

This section focuses on parameter estimation methods that can be combined with PFs. The basic methods are presented: the maximum likelihood (ML) method and the method with an augmented state vector (i.e., the state augmentation method).

### 2.2.1 Maximum likelihood method

There are two main ways of estimating model parameters: offline (batch) estimation, which is a method of estimating parameters for a specific period all at once, and online sequential estimation. The typical ML method is used for the former. First, the log-likelihood, which is a function of the parameter $\theta$, is expressed as follows:

$$\log p\left(y^{1:T}|\theta\right) = \sum_{n=1}^{T} \log p\left(y^n|y^{n-1},\theta\right). \tag{2.60}$$

By maximizing this likelihood, we can find the optimal parameters. Note that here it is assumed that the parameters do not change during the period, that is, they are static.

Typically, the maximization of the log-likelihood in Eq. (2.60) is carried out numerically, but this becomes difficult when the parameters become high-dimensional. Optimization methods such as the gradient descent method can be used if the likelihood is differentiable with respect to the parameters. In particular, with the recent development of machine learning and deep learning, many optimization methods have been developed and are summarized in [SCZZ19]. For example, there are first-order methods that include gradient descent, high-order methods such as Newton's method, and a derivative-free method. Generally, in learning system applications, optimization is often expressed as a problem of finding the parameter that minimizes (or maximizes) the objective function. Therefore, these methods can be applied by considering likelihood as the objective function. When the likelihood function is multimodal and local solutions exist, heuristic search methods, such as the greedy algorithm, or collective optimization, such as genetic algorithms [Hig97], can be considered. However, the computational cost is an issue for high-dimensional systems.

### 2.2.2 State augmentation method

This section describes an online sequential method using an augmented (or extended) state vector, namely, the state augmentation approach. The following equation defines the augmented state vector, including the state variables $x^n$ and the parameters $\theta^n$ at

time step $n$:

$$z^n = \begin{pmatrix} x^n \\ \theta^n \end{pmatrix}. \tag{2.61}$$

Then, the system model included in Eq. (2.2) can be extended as follows:

$$z^n = \begin{pmatrix} x^n \\ \theta^n \end{pmatrix} = \begin{pmatrix} f\left(x^{n-1}, \theta^{n-1}, \beta^n\right) \\ \theta^{n-1} + \eta^n \end{pmatrix}. \tag{2.62}$$

Here, the time evolution of the parameters is assumed to be a random walk, also known as artificial dynamics, and $\eta$ is a random parameter perturbation drawn from the pdf $\mathcal{N}(0, Q_\eta)$. Thus, as an estimation problem for the augmented state vector $z$, the parameters can be estimated simultaneously with the variables. Although applying PFs and other methods to estimate the state $z$ is possible, it has been pointed out that such a simple approach does not sufficiently explore the parameter space [Kit98].

The following assumes the application of PFs. If a random walk for parameters is assumed, as in Eq. (2.62), the parameter estimation performance also depends on the setting of the model-error covariance matrix $Q_\eta$. This becomes more pronounced in the estimation of time-varying parameters. If this variance is set large, it becomes easier to follow the parameter changes, but this may result in overfitting. In contrast, when the variance is small, the parameter distribution becomes narrower, that is, degeneracy occurs for the parameters, and parameter estimation may not be optimal. However, it is not always known in advance whether the parameters are time-varying. Note that the method of assuming a random walk for parameters is also used in static parameter estimation. In the case of general nonlinear and non-Gaussian state-space models, PF-based methods have difficulty even estimating time-invariant static parameters due to the degeneracy problem [ADT05]. In the past, due to this degeneracy problem, many methods were computationally inefficient, but for static parameters, computationally efficient methods are summarized in, for example, [KDS+15]. For example, the ML methods using gradient-based or expectation-maximization algorithms were cited as an online method that sequentially updates parameters as observations become available. This suggests that parameter estimation for data assimilation can be combined with algorithms developed primarily for machine learning, such as the gradient method.

## 2.3 Stochastic gradient descent

As described above, the ML method is used not only for batch (i.e., offline) estimation but also online estimation, as in [ADT05], [ADST04]. The key point is to incorporate the gradient descent method in the ML method into the stochastic approximation framework. This section summarizes the typical stochastic gradient method as a probabilistic approach that allows online estimation.

For example, particle filtering combined with a gradient algorithm and parameter estimation by recursive ML (RML) methods was proposed [DT03]. RML can also be described as a gradient-based method that maximizes the mean log-likelihood and is widely used in automatic control and signal processing [LS83]. Generally, a gradient descent method under a stochastic approximation is called a stochastic gradient descent (SGD) method. In SGD, the gradient computed from the entire data set is replaced by the gradient computed from a randomly selected data subset, which is considered a stochastic approximation of the gradient descent method. This method can iteratively optimize a differentiable objective function and has a lower computational load than that for the gradient descent method. While the gradient descent method is a batch (offline) estimation based on a data set, SGD is an online estimation method because it can be adapted to input data sequentially. Examples of the SGD method and its effect on large-scale learning problems are summarized in [Bot10].

The following is an example of a machine-learning framework. First, based on the loss function $l(\theta, x)$ obtained from the parameters to be estimated $\theta$ and the state $x$, and the pdf of the state $p(x)$, the expected loss $L(\theta)$ is defined as follows:

$$L(\theta) = \int l(\theta, x)\, p(x) dx. \tag{2.63}$$

Then, instead of the gradient of the expected loss $\nabla_\theta L(\theta)$, SGD uses its unbiased estimator, the stochastic gradient $\nabla_\theta l(\theta, x)$, to update parameters using the following formula:

$$\theta^t = \theta^{t-1} - \lambda \nabla_\theta l(\theta^{t-1}, x), \tag{2.64}$$

where $t$ is the number of iterations and $\lambda$ is the learning rate (step size factor). Eq. (2.64) shows that SGD only uses one gradient of the loss function, that is, only one dataset, for each parameter update (i.e., iteration). In the case of machine learning, using dif-

ferent data for each update can prevent convergence to a local solution and reduce computational complexity. Here, for the gradient of the loss function, the loss function must be differentiable, or the gradient must be calculated numerically. Note that [Fu15] summarizes a method for estimating stochastic gradients from stochastic simulations to optimize the parameters in a simulation model. Specifically, this includes finite difference methods (including simultaneous perturbation), perturbation analysis methods, likelihood ratio/score function methods, and the use of weak derivatives.

However, the SGD in Eq. (2.64) is sensitive to the learning rate or prone to overshoot, which can lead to a trade-off between convergence speed and convergence stability. Therefore, many improved algorithms have been proposed [SCZZ19]. Below is a brief introduction to the basic concepts of the moving-average (momentum) method, the method using root mean square propagation (RMSProp), and finally, adaptive moment estimation (Adam) [KB14], which is a combination of the two.

### Momentum method

In the typical momentum method, the parameters are updated with the following formula:

$$\begin{aligned} m^t &= \mu_m m^{t-1} + (1-\mu_m)\nabla_\theta l(\theta^{t-1}, x), \\ \theta^t &= \theta^{t-1} - \lambda m^t, \end{aligned} \tag{2.65}$$

where $m$ expresses the moving average (i.e., momentum) and $\mu_m$ is a hyperparameter. This means that the parameters are updated using a moving average of the gradient. A large amount of updating at one time can cause overshooting, which can cause the positive and negative values of the gradient to switch, forming so-called oscillations. In this momentum method, moving averages are used to suppress sudden fluctuations in gradient values.

### Root mean square propagation (RMSProp)

In the method using RMSProp, the parameters are updated as follows:

$$\begin{aligned} v^t &= \mu_v v^{t-1} + (1-\mu_v)\left\|\nabla_\theta l(\theta^{t-1}, x)\right\|^2, \\ \theta^t &= \theta^{t-1} - \frac{\lambda}{\sqrt{v^t + \epsilon_v}}\nabla_\theta l(\theta^{t-1}, x), \end{aligned} \tag{2.66}$$

where $v$ expresses the moving average for the squared gradient, $\mu_v$ is the hyperparameter, and $\epsilon_v$ is a small value that prevents zero division. Here, $\sqrt{v^t}$ represents the L2 norm of the gradient of the loss function based on the past gradient via $v^{t-1}$ term and the current gradient. This term scales the gradient, that is, it automatically adjusts the learning rate $\lambda$ according to the magnitude of the gradient.

### Adaptive moment estimation

Basically, the Adam method is a combination of momentum and RMSProp, and the parameters are updated as follows:

$$\begin{aligned} m^t &= \mu_m m^{t-1} + (1-\mu_m)\nabla_\theta l(\theta^{t-1}, x), \\ v^t &= \mu_v v^{t-1} + (1-\mu_v)\left\|\nabla_\theta l(\theta^{t-1}, x)\right\|^2, \\ \theta^t &= \theta^{t-1} - \lambda\frac{m^t}{\sqrt{v^t + \epsilon_v}}. \end{aligned} \tag{2.67}$$

Therefore, the Adam method combines the best parts of momentum and RMSProp. Because $\mu_m$ and $\mu_v$ are usually chosen to be close to one, the moving averages $m$ and $v$ are biased toward zero. Therefore, $\hat{m}$ and $\hat{v}$ are often used with the bias canceled in the following equation:

$$\hat{m}^t = \frac{m^t}{1-\mu_m}, \quad \hat{v}^t = \frac{v^t}{1-\mu_v}. \tag{2.68}$$

## 2.4 Summary

This chapter first presented an overview of PF-based methods with application to nonlinear high-dimensional models and parameter estimation. It introduced previous approaches to solving the problem of degeneracy, which is a particular challenge in high

dimensions. Then, the potential of IEWPF, which can equalize the weights of all particles, was presented. Next, we provided an overview of typical parameter estimation methods in state-space models and introduced optimization methods developed in machine learning in recent years. It was suggested that degeneracy is also an obstacle in parameter estimation. Based on a review of existing methods, we found that computationally efficient parameter estimation in nonlinear high-dimensional models with time-varying parameters is challenging. In the following chapters, we describe a time-varying parameter estimation method, which is an extension of IEWPF.

# 3

# Online state and time-varying parameter estimation using implicit equal-weights particle filter

This chapter describes an extension of the implicit equal-weights particle filter (IEWPF) [ZvLA16] to sequential parameter estimation. For resilient and quick estimation, we introduce an augmented state-space model with a correlated covariance matrix and a parameter nudging scheme inspired by an optimization algorithm in machine learning. This chapter is based on a previously published paper [SvLN24].

## 3.1 Introduction

Because parameters in numerical models are simplified representations of the modeled characteristics, parameter estimation plays an important role in obtaining accurate and reliable predictions. Also, the parameters can be considered time-varying as well as

static (see Section 1.1.2).

A typical method for time-varying state and parameter estimation in high-dimensional dynamical systems is the state augmentation technique, in which the parameter vector is incorporated into the state vector. This technique is also called joint estimation. According to [SJ15], the state augmentation method may become ineffective when the impact of parameters on the state is weak, and they propose a two-stage filter that combines a PF and an ensemble Kalman filter (EnKF). This method estimates the static parameters and tracks the dynamic variables alternatively. Although similar approaches using independent dual PFs [CP18] and a nested hybrid filter [PVMM18] have been proposed, they are only applicable to the estimation of static parameters. Extending the method to time-varying parameters requires identifying whether the change in observed states originates from state variables or parameters, but its utility in practical contexts depends on the cross-covariance between states and parameters. Particularly, detecting abrupt changes in characteristics in high-dimensional and partially observed nonlinear systems may be problematic because of the relatively low correlation between the observed state and parameters. Another issue concerns nonlinearities due to the temporal evolution of the system and augmented state vector. A parameter estimation method combined with a PF can deal with nonlinearities, but filter degeneracy might be a critical obstacle for high-dimensional systems such as geophysical and climate systems. To overcome this problem, several approaches have been proposed, including a hybrid PF and EnKF method [SJ15], as mentioned above. See Chapter 2 for other approaches.

In this chapter, we focus on a nonlinear time-varying system where the dimension of the state vector is large, while that of the model parameters is comparatively small, with a view to application in geophysical, climate, and other high-dimensional contexts. Then, we present a new PF-based parameter estimation method and assess the capability of detecting abrupt changes in characteristics by applying it to the above system. We provide a methodology and results based on IEWPF of [ZvLA16] as an example of avoiding filter degeneracy.

## 3.2 Methodology

### 3.2.1 Correlated perturbation in augmented state-space model

A typical state-space model for a nonlinear system containing model parameters is described as

$$\begin{aligned} x^n &= f\left(x^{n-1}, \theta^{n-1}\right) + \beta^n, \\ y^n &= h\left(x^n\right) + \epsilon^n, \end{aligned} \tag{3.1}$$

where $x^n$ is the state variable at time step $n$, $y^n$ is the observation vector at time step $n$, $f$ is a known possible nonlinear function that maps the state from time $t^{n-1}$ to $t^n$, $h$ is a known nonlinear observation operator, $\theta^n$ is the vector of model parameters whose true values are unknown and possibly time-varying, $\beta^n$ is a random model perturbation drawn from the model error probability distribution function (pdf) $\mathcal{N}(0, Q_\beta)$, and the observation error $\epsilon$ is drawn from the observation error pdf $\mathcal{N}(0, R)$. To estimate time-varying parameters sequentially, the state vector is updated according to the following dynamical system by augmenting parameters as artificial states:

$$\begin{pmatrix} x^n \\ \theta^n \end{pmatrix} = \begin{pmatrix} f\left(x^{n-1}, \theta^{n-1}\right) \\ \theta^{n-1} \end{pmatrix} + \begin{pmatrix} \beta^n \\ \eta^n \end{pmatrix}, \tag{3.2}$$

where $\eta^n$ is a random parameter perturbation drawn from the pdf $\mathcal{N}(0, Q_\eta)$ and we require $f$ to be a differentiable function with respect to the parameter. Then, the above state-updating function $f$ can be approximately expressed by a first-order Taylor series expansion for the previous parameter $\theta^{n-2}$:

$$f\left(x^{n-1}, \theta^{n-1}\right) \simeq f\left(x^{n-1}, \theta^{n-2}\right) + \left.\frac{\partial f}{\partial \theta}\right|_{\theta^{n-2}} \left(\theta^{n-1} - \theta^{n-2}\right). \tag{3.3}$$

This extracts the term that expresses the contribution of the parameter to the time evolution of the variable. This term is incorporated into the model error covariance matrix as the correlation between the variable and the parameter. By using the time evolution model in the previous time step $n-1$,

$$\theta^{n-1} = \theta^{n-2} + \eta^{n-1}, \tag{3.4}$$

we can rewrite Eq. (3.2) as

$$\begin{aligned} z^n &\equiv \begin{pmatrix} x^n \\ \theta^{n-1} \end{pmatrix} \\ &= \begin{pmatrix} f\left(x^{n-1}, \theta^{n-2}\right) \\ \theta^{n-2} \end{pmatrix} + \begin{pmatrix} \frac{\partial f}{\partial \theta}\Big|_{n-2} \eta^{n-1} + \beta^n \\ \eta^{n-1} \end{pmatrix} \\ &\equiv \tilde{f}\left(z^{n-1}\right) + \tilde{\rho}^n, \end{aligned} \tag{3.5}$$

where we introduce the augmented vector $z^n = [x^{nT}, \theta^{n-1^T}]^T$, model $\tilde{f}$, and perturbation $\tilde{\rho}^n$ representation. We also rewrite the observation operator $h$ in Eq. (3.1) as follows:

$$y^n = h_z\left(z^n\right) + \epsilon^n. \tag{3.6}$$

The augmented perturbation $\tilde{\rho}$ can be drawn from the error pdf $\mathcal{N}(0, \tilde{Q}^n)$, which is expressed as

$$\tilde{Q}^n = \begin{pmatrix} cov\left[\beta'^n, \beta'^n\right] & cov\left[\beta'^n, \eta^{n-1}\right] \\ \left(cov\left[\beta'^n, \eta^{n-1}\right]\right)^T & cov\left[\eta^{n-1}, \eta^{n-1}\right] \end{pmatrix}, \tag{3.7}$$

where $\beta'^n = \left(\partial f / \partial \theta\right) \eta^{n-1} + \beta^n$. Because model perturbation $\beta^n$ and parameter perturbation $\eta^{n-1}$ are independent of each other and both have zero means, each matrix element in Eq. (3.7) can be calculated as follows:

$$\begin{aligned} cov\left[\beta'^n, \beta'^n\right] &= E\left[\left(\frac{\partial f}{\partial \theta}\eta^{n-1} + \beta^n\right)\left(\frac{\partial f}{\partial \theta}\eta^{n-1} + \beta^n\right)^T\right] \\ &= E\left[\frac{\partial f}{\partial \theta}\eta^{n-1}\left(\eta^{n-1}\right)^T\left(\frac{\partial f}{\partial \theta}\right)^T + \beta^n\left(\beta^n\right)^T\right] \\ &= \frac{\partial f}{\partial \theta} Q_\eta^{n-1} \frac{\partial f}{\partial \theta}^T + Q_\beta^n, \end{aligned} \tag{3.8}$$

$$\begin{aligned} cov\left[\beta'^n, \eta^{n-1}\right] &= E\left[\left(\frac{\partial f}{\partial \theta}\eta^{n-1} + \beta^n\right)\left(\eta^{n-1}\right)^T\right] \\ &= E\left[\frac{\partial f}{\partial \theta}\eta^{n-1}\left(\eta^{n-1}\right)^T\right] \\ &= \frac{\partial f}{\partial \theta} Q_\eta^{n-1}, \end{aligned} \tag{3.9}$$

$$\begin{aligned} cov\left[\eta^{n-1}, \eta^{n-1}\right] &= E\left[\eta^{n-1}\left(\eta^{n-1}\right)^T\right] \\ &= Q_\eta^{n-1}. \end{aligned} \tag{3.10}$$

Then, Eq. (3.7) can be expressed as

$$\tilde{Q}^n = \begin{pmatrix} \frac{\partial f}{\partial \theta} Q_\eta^{n-1} \frac{\partial f}{\partial \theta}^T + Q_\beta^n & \frac{\partial f}{\partial \theta} Q_\eta^{n-1} \\ \left(\frac{\partial f}{\partial \theta} Q_\eta^{n-1}\right)^T & Q_\eta^{n-1} \end{pmatrix}. \tag{3.11}$$

Note that the Taylor expansion in Eq. (3.3) is used up to the first-order term, so the augmented perturbation $\tilde{\rho}$ from $\tilde{Q}$ includes the linear impact of the parameters on the model evolution over one time step.

### 3.2.2 State and parameter update with IEWPF

In this section, we explain how to apply IEWPF to update equation Eq. (3.5) and how to avoid filter degeneracy. When a Markovian system has observational errors that are independent from one time to another, the forecast pdf can be written as

$$p\left(z^n | y^{1:n-1}\right) = \int p\left(z^n | z^{n-1}\right) p\left(z^{n-1} | y^{1:n-1}\right) dz^{n-1}. \tag{3.12}$$

Then, after Eq. (3.12) is plugged into Bayes' theorem as a prior pdf, the posterior pdf for the model state given observations can be written as

$$p\left(z^n | y^{1:n}\right) = \frac{p\left(y^n | z^n\right)}{p(y^n | y^{1:n-1})} \int p\left(z^n | z^{n-1}\right) p\left(z^{n-1} | y^{1:n-1}\right) dz^{n-1}. \tag{3.13}$$

Suppose we run a PF, and the particle weight for the ensemble at the previous time step $n-1$ is given by

$$p\left(z^{n-1} | y^{1:n-1}\right) = \frac{1}{N} \sum_{i=1}^{N} \delta\left(z^{n-1} - z_i^{n-1}\right). \tag{3.14}$$

Then, by plugging Eq. (3.14) into Eq. (3.13), we can obtain

$$p\left(z^n | y^{1:n}\right) = \frac{1}{N} \sum_{i=1}^{N} \frac{p(y^n | z^n) p(z^n | z_i^{n-1})}{p(y^n | y^{1:n-1})}. \tag{3.15}$$

Introducing the proposal density $q(z^n|z_{1:N}^{n-1}, y^n)$ that is conditioned on all particles at time $n-1$, indicated by the $z_{1:N}^{n-1}$, we can express Eq. (3.15) as

$$p\left(z^n|y^{1:n}\right) = \frac{1}{N}\sum_{i=1}^{N}\frac{p(y^n|z^n)p(z^n|z_i^{n-1})}{p(y^n|y^{1:n-1})q(z^n|z_{1:N}^{n-1}, y^n)}q(z^n|z_{1:N}^{n-1}, y^n). \tag{3.16}$$

The well-known problem of filter degeneracy means that the weights are concentrated on only some particles, and most particles have a negligible weight after a few propagations. [SBM15] reports that a PF using the optimal proposal yields minimal degeneracy and provides performance bounds. This could be a serious obstacle to implementing a PF when the number of states and observations increase, as in a high-dimensional system. Therefore, we use IEWPF [ZvLA16], which can avoid this filter degeneracy problem. From Eq. (3.14), Eq. (3.16) can be expressed as

$$p\left(z^n|y^{1:n}\right) = \frac{1}{N}\sum_{i=1}^{N} w_i^n \delta\left(z^n - z_i^n\right), \tag{3.17}$$

where $w_i^n$ is the weight for particle $i$ and is expressed as follows using the proposal density given in Eq. (3.16):

$$w_i^n = \frac{p(y^n|z_i^n)}{p(y^n|y^{1:n-1})}\frac{p(z_i^n|z_i^{n-1})}{q(z_i^n|z_{1:N}^{n-1}, y^n)}. \tag{3.18}$$

Instead of drawing directly from the proposal density $q$, we can draw particles from a standard Gaussian distribution proposal density $q(\xi)$, which is related by

$$q(\xi^n) = q(z^n|z_{1:N}^{n-1}, y^n)\left\|\frac{dz^n}{d\xi^n}\right\|, \tag{3.19}$$

where $\|dz/d\xi\|$ denotes the absolute value of the determinant of the Jacobian matrix, which expresses the following transformation:

$$z_i^n = \zeta_i^n + \alpha_i^{1/2} P^{1/2} \xi_i^n, \tag{3.20}$$

where $\zeta_i^n$ represents the mode of $q(z^n|z_{1:N}^{n-1}, y^n)$, $P$ is a measure of the width of that pdf, and $\alpha_i$ is a scalar factor. The specific method of giving the matrix $P$ is described below. Note that this expression is similar to the original IEWPF [ZvLA16], but $z_i^n$ denotes

the augmented vector $z^n = [x^{nT}, \theta^{n-1^T}]^T$. This means that the transformed variable $\xi$ also has the dimension of the augmented vector. Then, Eq. (3.18) can be expressed as follows:

$$w_i^n = \frac{p(y^n|z_i^n)}{p(y^n|y^{1:n-1})} \frac{p(z_i^n|z_i^{n-1})}{q(\xi_i^n)} \left\| \frac{dz_i^n}{d\xi_i^n} \right\|. \tag{3.21}$$

In general, $\zeta_i^n$ can be obtained via a minimization of $-\log q\left(z^n|z_{1:N}^{n-1}, y^n\right)$, similar to, for example, a three-dimensional variational scheme, and also the equal weights can be obtained numerically. In this chapter, we follow [ZvLA16] and assume a linear observation operator, which allows for an analytical solution for equal weights.

### 3.2.3 Linear observation model and Gaussian error

In the following discussion, we assume the linear observation operator $\tilde{H}$ and the Gaussian model and observation error as given in Eq. (3.5) and Eq. (3.6).Then, by choosing the proposal density as the optimal proposal density, that is, $q(z^n|z_{1:N}^{n-1}, y^n) = p(z^n|z_i^{n-1}, y^n)$, $\zeta_i^n$ in Eq. (3.20) can be expressed as follows:

$$\zeta_i^n = \tilde{f}\left(z_i^{n-1}\right) + K\left\{y^n - \tilde{H}\tilde{f}\left(z_i^{n-1}\right)\right\}, \tag{3.22}$$

where

$$K = \tilde{Q}\tilde{H}^T\left(\tilde{H}\tilde{Q}\tilde{H}^T + R\right)^{-1}. \tag{3.23}$$

Also, in this case, the matrix $P$ in Eq. (3.20) is analytically given by

$$P = \left(\tilde{Q}^{-1} + \tilde{H}^T R^{-1}\tilde{H}\right)^{-1}, \tag{3.24}$$

where $\tilde{Q}$ is the model error covariance matrix described in Eq. (3.11) and $R$ is the observation error covariance matrix. Therefore, from Eq. (3.20) and Eq. (3.22), equal-weight particle $z_i$ sampled from posterior pdf Eq. (3.16) can be constructed using the scalar factor $\alpha_i$. Note that the matrix $P$ can also be given by an analysis covariance matrix, for example, as obtained from the ensemble perturbations used in an ensemble transform Kalman filter (ETKF) [BEM01].

The factor $\alpha_i$ needs to be determined so that the weight of each particle $i$ represented by Eq. (3.21) is the same target weight for all particles. The update from the

weight $w_i^{n-1}$ of the previous time step is expressed from Eq. (3.21) as

$$w_i^n = w_i^{n-1} \frac{p(y^n|z_i^n)p(z_i^n|z_i^{n-1})}{q(\xi_i^n)} \left\| \frac{dz_i^n}{d\xi_i^n} \right\|. \tag{3.25}$$

With the Gaussian assumption, we can write

$$\begin{aligned} &p(y^n|z^n)p(z^n|z_i^{n-1}) \propto \\ &\exp\left[-\frac{1}{2}\left(y^n - \tilde{H}z^n\right)^T R^{-1}\left(y^n - \tilde{H}z^n\right) - \frac{1}{2}\left(z^n - \tilde{f}\left(z_i^{n-1}\right)\right)^T \tilde{Q}^{-1}\left(z^n - \tilde{f}\left(z_i^{n-1}\right)\right)\right] \\ &= \exp\left[-\frac{1}{2}\left(z^n - z_i^a\right)^T P^{-1}\left(z^n - z_i^a\right)\right] \exp\left(-\frac{1}{2}\phi_i\right), \end{aligned} \tag{3.26}$$

where

$$\phi_i = \left[y^n - \tilde{H}\tilde{f}\left(z_i^{n-1}\right)\right]^T \left(\tilde{H}\tilde{Q}\tilde{H}^T + R\right)^{-1} \left[y^n - \tilde{H}\tilde{f}\left(z_i^{n-1}\right)\right]. \tag{3.27}$$

Taking the logarithm of Eq. (3.25) leads to

$$-2\log w_i^n = -2\log w_i^{n-1} + \left[-2\log\left(\frac{p(y^n|z_i^n)p(z_i^n|z_i^{n-1})}{q(\xi_i^n)} \left\| \frac{dz_i^n}{d\xi_i^n} \right\|\right)\right]. \tag{3.28}$$

Substituting Eq. (3.26) and Eq. (3.20) into Eq. (3.28), we find

$$-2\log w_i^n = -2\log w_i^{n-1} + \alpha_i \xi_i^{nT} P^{1/2} P^{-1} P^{1/2} \xi_i^n + \phi_i - \xi_i^{nT}\xi_i^n - 2\log\left(\left\| \frac{dz_i^n}{d\xi_i^n} \right\|\right). \tag{3.29}$$

Using Eq. (3.20) and the simplified expression for the Jacobian in [ZvLA16] Eq. (19), we can rewrite the above equation as

$$\begin{aligned} -2\log w_i^n &= -2\log w_i^{n-1} + (\alpha_i - 1)\,\xi_i^{nT}\xi_i^n + \phi_i \\ &\quad - 2\log\left(\alpha_i^{N_x/2} \left\|P^{1/2}\right\| \left|1 + \frac{\partial \alpha_i^{1/2}}{\partial \xi_i^n} \frac{\xi_i^n}{\alpha_i^{1/2}}\right|\right) \\ &= -2\log w_i^{n-1} + (\alpha_i - 1)\,\xi_i^{nT}\xi_i^n + \phi_i - 2N_x \log \alpha_i^{1/2} \\ &\quad - 2\log\left(\left\|P^{1/2}\right\|\right) - 2\log\left(\left|1 + \frac{\partial \alpha_i^{1/2}}{\partial \xi_i^{nT}} \frac{\xi_i^n}{\alpha_i^{1/2}}\right|\right), \end{aligned} \tag{3.30}$$

where $N_x$ is the dimension of the model state. Setting the weights for all particles to the target weight is equivalent to setting all $\log w_i$ equal to the constant $-\frac{C}{2} + \log\left(\left\|P^{1/2}\right\|\right)$, which leads to the following equation for $\alpha_i$:

$$(\alpha_i - 1)\,\xi_i^{nT}\xi_i^n - 2N_x \log \alpha_i^{1/2} - 2\log\left(\left|1 + \frac{\partial \alpha_i^{1/2}}{\partial \xi_i^{nT}} \frac{\xi_i^n}{\alpha_i^{1/2}}\right|\right) = C - \left(\phi_i - 2\log w_i^{n-1}\right), \tag{3.31}$$

in which the constant value $2\log\left(\left\|P^{1/2}\right\|\right)$ is included in $C$. Here, let $c_i$ denote the log-weight offsets for each particle $i$ from the target weight $C$ as

$$c_i = C - \left(\phi_i - 2\log w_i^{n-1}\right). \tag{3.32}$$

In practice, this $c_i$ can be determined using the values of $\phi$ for all particles as

$$c_i = \max_j\{\phi_j\} - \phi_i. \tag{3.33}$$

Therefore, $\alpha_i$ is obtained as a solution satisfying Eq. (3.31) with $c_i$ determined by Eq. (3.33).

Further assuming that the factor $\alpha_i$ depends on $\xi_i^n$ only through $g_i = \xi_i^{nT}\xi_i^n$, we can simplify Eq. (3.31) to

$$\exp\left(-\frac{\alpha_i g_i}{2}\right)(\alpha_i g_i)^{N_x/2-1}\left\|\frac{d\left(\alpha_i g_i\right)}{dg_i}\right\| = \exp\left(-\frac{g_i}{2}\right)g_i^{N_x/2-1}\exp\left(-\frac{c_i}{2}\right), \tag{3.34}$$

(see the Appendix in [ZvLA16]). For every particle to reach the target weight, $c_i \geq 0$ should be satisfied; therefore, $0 < \exp\left(-c_i/2\right) \leq 1$ in Eq. (3.34). Furthermore, because the function $\exp\left(-\alpha g_i/2\right)(\alpha g_i)^{N_x/2-1}$ on the left side has an extremum at $\alpha_i = (N_x - 2)/g_i$, it is suggested that the solution $\alpha_i$ of Eq. (3.34) allows two values. According to [ZvLA16], Eq. (3.34) can be integrated from $N_x/2$ to $\infty$ to yield the following equation:

$$\Gamma\left(\frac{N_x}{2}, \frac{\alpha_i g_i}{2}\right) = \begin{cases} \exp\left(-\frac{c_i}{2}\right)\Gamma\left(\frac{N_x}{2}, \frac{g_i}{2}\right) & \text{if } \frac{d\left(\alpha_i g_i\right)}{dg_i} > 0\,, \\ \exp\left(-\frac{c_i}{2}\right)\gamma\left(\frac{N_x}{2}, \frac{g_i}{2}\right) & \text{if } \frac{d\left(\alpha_i g_i\right)}{dg_i} < 0\,, \end{cases} \tag{3.35}$$

where $\Gamma$ is a monotonically decreasing upper incomplete gamma function, while $\gamma$ is a monotonically increasing lower incomplete gamma function. Figure 3.1 shows examples of functions on the left and right-hand side of Eq. (3.35) under the numerical experimental conditions (i.e., $N_x = 1000$) in Section 3.3.1. Note that because $N_x/2 = 500$, the neighborhood of $g_i/2 = 500$ are drawn. For the log-weight offsets $c_i$ of each particle, the intersection of graph (a), representing the left-hand side of Eq. (3.35), and graph (b), representing the right-hand side, is the solution $\alpha_i$. Therefore, the solution $\alpha_i$ for every particle $i$ that satisfies Eq. (3.35) is allowed to be both $\alpha \leq 1$ and $\alpha \geq 1$. Note

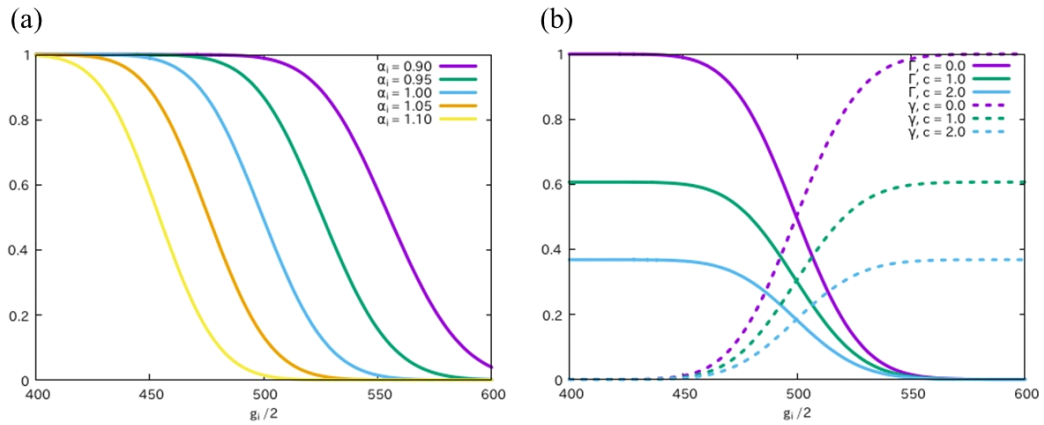


Figure 3.1: Examples of the functions shown in Eq. (3.35) for the numerical experimental conditions ($N_x = 1000$) in Section 3.3.1: (a) the left-hand side of Eq. (3.35) when $\alpha_i$ is set to 0.90, 0.95, 1.00, 1.05, 1.10, (b) in the right-hand side of Eq. (3.35) when $c_i = 0$, 1.0, and 2.0, the upper incomplete gamma function ($\Gamma$) and the lower incomplete gamma function ($\gamma$).

that in the following calculation examples, $\alpha_i$ is not obtained from the intersection, but an approximate expression under higher dimensions is used. Although $\alpha \geq 1$ solutions are known to lead to systematic bias [ZvLA16], the bias decreases when the state-space dimension $N_x$ increases in high-dimensional cases. As another solution, [SEvLA19] proposed a two-stage IEWPF that can eliminate this bias.

In practice, the following should be considered when generating the posterior distribution by calculating $\alpha_i$ that satisfies Eq. (3.35). The first point is the computational cost of finding $\alpha_i$ numerically for each particle. To avoid this calculation, [ZvLA16] proposed an approximation under the limiting case of $N_x \to \infty$. Then, the solution $\alpha$ can be expressed analytically using the Lambert W function [CGH+96], as expressed in Eq. (2.55). This function has two branches: $\alpha > 1$, which gives a large ensemble spread, and $\alpha < 1$, which gives the opposite effect. Therefore, it is proposed to adjust the ratio of sampling $\alpha_i$ for each particle $i$ from either branch to bring the shape of the distribution closer to the ideal one. The results for the $\alpha$ dependency are shown later. The second point is the guarantee of convergence to the posterior distribution. IEWPF can equalize the weights of all particles, but the convergence of the filter distribution to the posterior distribution was only confirmed experimentally by [ZvLA16] and not

shown theoretically.

### 3.2.4 Parameter nudging with proposal density

The effectiveness of the method proposed in the previous section, which augments parameters as artificial states, depends on the cross-covariance between states and parameters. To improve the accuracy and resilience of time-varying parameters, we introduce an optimization algorithm from machine learning into the parameter time evolution model using the flexibility of the proposal density in particle filtering. Using the model error covariance matrix $\tilde{Q}$ in Eq. (3.11), the model transition density is expressed as

$$p\left(z^n|z^{n-1}\right) = \mathcal{N}\left(\tilde{f}\left(z^{n-1}\right), \tilde{Q}^n\right). \tag{3.36}$$

The forecast pdf expressed in Eq. (3.12) is allowed to both divide and multiply the model transition density by a proposal transition density $q$, leading to

$$p\left(z^n|y^{1:n-1}\right) = \int \frac{p\left(z^n|z^{n-1}\right)}{q\left(z^n|z_{1:N}^{n-1}, y^n\right)} q\left(z^n|z_{1:N}^{n-1}, y^n\right) p\left(z^{n-1}|y^{1:n-1}\right) dz^{n-1}. \tag{3.37}$$

Drawing from $p\left(z^n|z^{n-1}\right)$ corresponds to using the original model transition density Eq. (3.36). Still, we could instead draw from $q\left(z^n|z_{1:N}^{n-1}, y^n\right)$, which would correspond to any other model transition that we choose. This allows us to control the transition of both state and parameters by choosing proposal density $q$.

Sequential observation data can be considered as samples for the stochastic gradient descent (SGD) algorithm based on the similarity between sequential data assimilation and online learning or stochastic optimization in that the data are given sequentially. The ideas in stochastic optimization have advanced in recent years in machine learning and deep learning with large-scale data. The basic problem structure classification and associated solutions are summarized in [Han15]. The effectiveness of SGD for large-scale learning problems, that is, cases with large-scale data, is also described in [Bot10]. The optimization algorithm used in the present IEWPF method is described in the next section. We assume an objective function $L_i^n(\theta)$ and consider the problem of minimizing this function. The parameter $\theta^n$ can be updated by the following iteration:

$$\theta^n \leftarrow \theta^{n-1} - \lambda g^n, \quad \left(e.g., g^n \in \nabla L_i^n(\theta)\right), \tag{3.38}$$

where $\lambda$ is the step size, sometimes called the learning rate in machine learning contexts. The function $g^n$ expresses the update rule for the parameter.

Here, we consider introducing the above parameter update analogy to the transition density modification. In the next step of the last observation $n$, that is, $n+1$, we give the proposal density $q$ at time step $n+1$ for augmented state $z$ as

$$\begin{aligned} q\left(z_i^{n+1}|z_i^n, y^n\right) &= \mathcal{N}\left(\tilde{f}\left(z_i^n\right) + \begin{pmatrix} 0 \\ -\lambda g\left(\theta_i^{n-1}, y^n\right) \end{pmatrix}, \tilde{Q}^{n+1}\right) \\ &\equiv \mathcal{N}\left(\tilde{f}\left(z_i^n\right) + \tilde{g}^n, \ \tilde{Q}^{n+1}\right), \end{aligned} \tag{3.39}$$

where the augmented nudging term is denoted as $\tilde{g}^n$. Therefore, the step size $\lambda$ and the function $g(\theta_i^{n-1}, y^n)$ have the same role as Eq. (3.38) and together express the nudging term, forcing estimated model parameters towards true values, and $y^n$ is the last observed data vector. $\tilde{Q}^{n+1}$ is the same augmented model error covariance matrix as described in Eq. (3.11) with correlated perturbation. Then, updating the augmented state vector after the last observation step $n$ is given as follows, instead of the original updating expressed in Eq. (3.5):

$$z_i^{n+1} = \tilde{f}\left(z_i^n\right) + \hat{\rho}_i^{n+1}, \tag{3.40}$$

where

$$p\left(\hat{\rho}^{n+1}\right) = \mathcal{N}\left(\tilde{g}^n, \ \tilde{Q}^{n+1}\right). \tag{3.41}$$

This corresponds to only modifying the augmented perturbation $\hat{\rho}^{n+1}$, which shifts the mean value of the parameters. Note that sampling from this proposal transition density instead of the original model is compensated by an extra weight as described in [AvL15]:

$$\begin{aligned} &\frac{p\left(z_i^{n+1}|z_i^n\right)}{q\left(z_i^{n+1}|z_i^n, y^n\right)} \propto \exp\left[ -\frac{1}{2}\left(z_i^{n+1} - \tilde{f}(z_i^n)\right)^T \tilde{Q}^{-1}\left(z_i^{n+1} - \tilde{f}(z_i^n)\right)\right. \\ &\left. + \frac{1}{2}\left(z_i^{n+1} - \left(\tilde{f}(z_i^n) + \tilde{g}^n\right)\right)^T \tilde{Q}^{-1}\left(z_i^{n+1} - \left(\tilde{f}(z_i^n) + \tilde{g}^n\right)\right)\right] \\ &= \exp\left[ -\frac{1}{2}\left(\hat{\rho}_i^{n+1}\right)^T \tilde{Q}^{-1}\ \hat{\rho}_i^{n+1} + \frac{1}{2}\left(\hat{\rho}_i^{n+1} - \tilde{g}^n\right)^T \tilde{Q}^{-1}\left(\hat{\rho}_i^{n+1} - \tilde{g}^n\right)\right]. \end{aligned} \tag{3.42}$$

### 3.2.5 Adam method-based parameter nudging

As discussed above, we introduced a nudging term for the parameters by taking advantage of the flexibility of the proposal density in particle filtering. One of the main points of this thesis is that we can choose any term that forces the parameters toward the true value. Therefore, our scheme is combined with a well-known gradient descent optimization algorithm that has evolved in recent years as deep learning progresses [ATY+18]. In general, a task in machine learning and deep learning is often expressed as the problem of finding parameters that minimize (or maximize) the objective function, and the key is to quickly find the optimal parameters. Typical optimization formulations and algorithms are summarized in [SCZZ19].

Regarding gradient-based optimization algorithms, [Rud16] showed a classification of algorithms and a description of typical examples. Momentum-based algorithms accumulate a decaying sum of the previous gradients into a momentum vector and use that instead of the true gradients. This method has the advantage of accelerating optimization along dimensions where the gradient remains relatively consistent and slowing it along turbulent dimensions where the gradient is significantly oscillating. Another approach is norm-based algorithms, which divide a portion of the gradient by the $L_2$ norm of all previous gradients. This has the advantage of slowing down optimization along dimensions that have already changed and accelerating it along dimensions that have only changed slightly. In our method, we use the adaptive moment estimation (Adam) proposed by [KB14], which combines the above two approaches.

Our proposed formulation of the function $g(\theta_i^{n-1}, y^n)$ for the parameter nudging term in Eq. (3.39) is as follows. First, $\tilde{f}(z_i^{n-1})$ can be regarded as the expected value of $z_i^n$ given $z_i^{n-1}$ and is defined by

$$\bar{z}_i^n = E[z_i^n | z_i^{n-1}] = \tilde{f}(z_i^{n-1}). \tag{3.43}$$

Next, we chose the log-likelihood of $p\left(y^n | \bar{z}_i^n\right)$ as the aforementioned objective function $L_i^n$ in Eq. (3.38) as follows:

$$L_i^n \equiv -2 \log \left[ p\left(y^n | \bar{z}_i^n\right) \right]. \tag{3.44}$$

Here, Eq. (3.44) can be calculated from the likelihood with respect to the observed

value $y^n$ at observation step $n$ and ensemble member $i$, given $\bar{z}_i^n$, as follows:

$$p\left(y^n|\bar{z}_i^n\right) \propto \exp\left[-\frac{1}{2}\left(y^n - \tilde{H}\bar{z}_i^n\right)^T R^{-1}\left(y^n - \tilde{H}\bar{z}_i^n\right)\right]. \tag{3.45}$$

Then, we define the function $g(\theta_i^{n-1}, y^n)$ in Eq. (3.39) by using the gradient of the objective function $L_i^n$ as follows. Following [KB14], we introduce the moving averages of the gradient and the squared gradient, and denote them as $m_i^n$ and $v_i^n$, respectively. Their update equations are expressed using the gradient of $L_i^n$ as follows:

$$\begin{aligned} m_i^n &= \mu_m m_i^{n-1} + (1-\mu_m)\nabla_\theta L_i^n, \\ v_i^n &= \mu_v v_i^{n-1} + (1-\mu_v)\left\|\nabla_\theta L_i^n\right\|^2, \end{aligned} \tag{3.46}$$

where the hyperparameters $\mu_m$ and $\mu_v$ control the decay rate for these moving averages. Note that the gradient $\nabla_\theta L_i^n$ requires computing the partial derivatives of the likelihood with respect to the parameters in Eq. (3.45). Because these moving averages are initialized (as a vector of zeros), the moment estimates are biased toward zero, especially during the initial time step and especially when the decay rates are low (i.e., $\mu_m$ and $\mu_v$ are chosen to be close to 1). According to [KB14], $m_i^n$ and $v_i^n$ in Eq. (3.46) are modified as follows to cancel these biases:

$$\hat{m}_i^n = \frac{m_i^n}{1-\mu_m}, \; \hat{v}_i^n = \frac{v_i^n}{1-\mu_v}. \tag{3.47}$$

Finally, the function $g(\theta_i^{n-1}, y^n)$ expressed in Eq. (3.39) is obtained as follows:

$$g(\theta_i^{n-1}, y^n) = \frac{\hat{m}_i^n}{\sqrt{\hat{v}_i^n} + \delta}. \tag{3.48}$$

Here, the factor $\sqrt{\hat{v}_i^n}$ represents the $L_2$ norm of the past gradients via the $v_i^{n-1}$ term and current gradient in Eq. (3.46), and scales the gradient. Note that $\delta$ is a factor to prevent dividing by zero and is set to $1.0 \times 10^{-8}$ in the following experiment.

The present method contains two procedures dependent on the observation: (1) state and parameter update by the IEWPF and computation of the likelihood gradient at the observation step, and (2) parameter nudging with proposal density between observations. The algorithm is summarized as follows:

**(1) State and parameter update at the observation step**

- Sample initial particle for state $x_i^0$ and parameter $\theta_i^0, i = 1, \ldots, N$.
- For every model time step $k$:
  - Perform a forecast based on model transition.
  - Differentiate the state-updating function $f$ by the parameter $\theta$, and then update the augmented covariance matrix $\tilde{Q}^k$.
- When the model reaches the observation time $n$, for each particle $i$:
  - Compute $\phi_i$ for all particles with Eq. (3.27), and then determine $c_i$ from Eq. (3.33).
  - Calculate $\alpha_i$ that satisfies Eq. (3.35) using the analytical solution of the Lambert W function expressed as Eq. (2.55).
  - Update the state and parameter using Eq. (3.20) and Eq. (3.22).
  - Normalize and update the weight.
- In preparation for the next forecast step:
  - Compute likelihood $p\left(y^n|\bar{z}_i^n\right)$ from observation $y^n$ and observation error covariance $R$ with Eq. (3.45), and then obtain likelihood gradient $\nabla_\theta L_i^n$ from Eq. (3.44).
  - Compute parameter nudging term $\lambda g\left(\theta_i^{n-1}, y^n\right)$ from Eq. (3.48), by using hyperparameters $\mu_m$ and $\mu_v$ and step-size factor $\lambda$.

**(2) Parameter nudging at the forecast step**

- At the time step $n+1$ in the next step after observation, for each particle $i$:
  - Generate parameter perturbation using the computed parameter nudging term $\lambda g\left(\theta_i^{n-1}, y^n\right)$ from Eq. (3.41).
  - Compute the extra weight in Eq. (3.42).
  - Perform a forecast using Eq. (3.40).

## 3.3 Numerical experiments

The effectiveness of the proposed method was demonstrated through two synthetic test cases as follows. The first case is a linear model with additive parameters, where all model states are observed directly at every time step. Although this thesis focuses on a nonlinear system, we use a linear model to verify that the shape of the posterior pdf is close to the true one. The second case is the Lorenz 96 model ([Lor96]) with parameterized forcing, where only the model states are observed directly at every fourth step.

### 3.3.1 Linear model with unknown parameter

To compare the estimates of the present method with the analytically calculated true values, we use the following linear model:

$$\begin{pmatrix} x^n \\ \theta^n \end{pmatrix} = \begin{pmatrix} F_x & F_{x\theta} \\ O & I \end{pmatrix} \begin{pmatrix} x^{n-1} \\ \theta^{n-1} \end{pmatrix} + \begin{pmatrix} \beta^n \\ \eta^n \end{pmatrix}, \tag{3.49}$$

where $x \in \mathbb{R}^{N_x}$ is a model state vector with dimension $N_x$ and $\theta \in \mathbb{R}^{N_\theta}$ is a parameter vector with dimension $N_\theta$. $\beta^n$ and $\eta^n$ are random perturbations drawn from the model error pdf $\mathcal{N}(0, Q_\beta)$ and parameter error pdf $\mathcal{N}(0, Q_\eta)$, respectively. The matrices $F_x \in \mathbb{R}^{N_x \times N_x}$ and $F_{x\theta} \in \mathbb{R}^{N_x \times N_\theta}$ represent the linear model. Here, we define the following matrices:

$$\tilde{F} = \begin{pmatrix} F_x & F_{x\theta} \\ O & I \end{pmatrix}, \quad \tilde{G} = \begin{pmatrix} I & F_{x\theta} \\ O & I \end{pmatrix}. \tag{3.50}$$

Then, Eq. (3.49) can be rewritten by using Eq. (3.4) as follows:

$$\begin{pmatrix} x^n \\ \theta^{n-1} \end{pmatrix} = \tilde{F} \begin{pmatrix} x^{n-1} \\ \theta^{n-2} \end{pmatrix} + \tilde{G} \begin{pmatrix} \beta^n \\ \eta^{n-1} \end{pmatrix}, \tag{3.51}$$

When the initial prior pdf is Gaussian, the true posterior pdf should also be Gaussian. If the posterior pdf at time $n-1$ is Gaussian with covariance matrix $P_{n-1|n-1}$, the predicted covariance matrix $P_{n|n-1}$ of the prior pdf expressed in Eq. (3.51) can be calculated as follows:

$$P_{n|n-1} = \tilde{F} P_{n-1|n-1} \tilde{F}^T + \tilde{G} \tilde{Q} \tilde{G}^T, \tag{3.52}$$

where

$$\tilde{G}\tilde{Q}\tilde{G}^T = \begin{pmatrix} F_{x\theta}Q_\eta F_{x\theta}^T + Q_\beta & F_{x\theta}Q_\eta \\ \left(F_{x\theta}Q_\eta\right)^T & Q_\eta \end{pmatrix}, \tag{3.53}$$

and this term is equivalent to Eq. (3.11) when using the linear model $\tilde{F}$ defined in Eq. (3.50) and Eq. (3.51).

In the following experiments, we choose the dimension of the model state $N_x = 1000$ and the parameter $N_\theta = 1$ to consider a simple high-dimensional system with one parameter. To set the model, $F_x = I$, $F_{x\theta} = 0.1$, and the time evolution model described in Eq. (3.51) and observation model are expressed as

$$\begin{aligned} z^n &= \begin{pmatrix} x_j^n \\ \theta^{n-1} \end{pmatrix} = \begin{pmatrix} x_j^{n-1} + 0.1\,\theta^{n-2} \\ \theta^{n-2} \end{pmatrix} + \begin{pmatrix} \beta^n + 0.1\eta^{n-1} \\ \eta^{n-1} \end{pmatrix}, \\ y^n &= \tilde{H}z^n + \epsilon^n, \end{aligned} \tag{3.54}$$

where index $j = 1, \ldots, N_x$ indicates the elements of the model state $x$. Here, the observation model $\tilde{H} = \begin{pmatrix} I & 0 \end{pmatrix}$ when all variables are observed, and $\epsilon$ is the observation error drawn from the observation error pdf $\mathcal{N}(0, R)$. Because we assume a time-independent state transition matrix $\tilde{F}$, the covariance matrix satisfying the linear system defined by Eq. (3.54) converges to the steady-state matrix $P$ such that $P_{n|n-1} = P_{n-1|n-2} \equiv P$, and satisfies the discrete-time Riccati equation [Won68] as follows:

$$P = \tilde{F}P\tilde{F}^T - \tilde{F}P\tilde{H}^T\left(\tilde{H}P\tilde{H}^T + R\right)^{-1}\tilde{H}P\tilde{F}^T + \tilde{G}\tilde{Q}\tilde{G}^T. \tag{3.55}$$

Therefore, the shape of the true posterior pdf in Eq. (3.54) can be obtained by solving Eq. (3.55) numerically and comparing to the distribution obtained from our IEWPF.

The procedure for the comparison using synthetic data is as follows. Let us assume the initial ensemble members $z_i^0$ are sampled from the background error $\mathcal{N}(0, B)$. First, one member from the ensemble generated under the model error covariance matrix $Q$ and the background error covariance matrix $B$ is used as the "truth." Observations are then created from this "truth" and the observation error defined by covariance matrix $R$. In the following experiments, the true value of the parameter is 0. The background error covariance matrix for the states $B_x$ and the parameter $B_\theta$ are chosen as a diagonal matrix with the main diagonal values of 1 and 0, respectively. The true model error covariance matrix for the state $Q_\beta$ and the parameter $Q_\eta$ are chosen as a diagonal matrix

with the main diagonal values of 0.04 and 0, respectively. The observation error matrix $R$ is diagonal, and the main diagonal value is set to 0.01.

Next, for the assimilation, we choose the same matrices $Q_\beta$, $B_x$, and $R$ as when the observation was generated. The main diagonal values for matrices $Q_\eta$ and $B_\theta$ for parameters are set to be 0.04 and 0.001, respectively. The number of particles is set to $N = 20$ to demonstrate the validity of the estimation with a sufficiently small number of particles compared to the dimension of the state (i.e., $N_x = 1000$). In the experiment, all model state variables $x$ are observed at every step. Note that the step size $\lambda$ in Eq. (3.39) is set to zero to evaluate the parameter augmentation method of IEWPF described in Section 3.2.2. To investigate the dependency of aforementioned $\alpha_i$ on the shape of the posterior pdf, we compare the variance of pdfs estimated with the values sampled from the $\alpha_i \geq 1$ branch at three sampling percentages: 0%, 50%, and 100%. Note that 50% means sampling from both branches of $\alpha_i \geq 1$ and $\alpha_i \leq 1$, which is the closest to the true pdf according to [ZvLA16]. Thus, 0% and 100% mean sampling only from the $\alpha_i \leq 1$ branch and $\alpha_i \geq 1$ branch, respectively.

Figure 3.2 shows an example of the calculated $\alpha_i$ from the Lambert W function, which is the approximate analytical solution of Eq. (3.35). The calculated $\alpha_i$ is the accumulation of 1000th steps with 20 particles, showing both $W_0$ and $W_{-1}$ branches. The analytical solutions $\alpha_i(g_i)$ as a function of $g_i$ with different log-weight offsets $c_i$ shown as Figure 2.3 in Chapter 2 are also drawn. From this figure, one can read off the log-weight offset $c_i$ corresponding to the calculated $\alpha_i$.

Figure 3.3 shows histograms of the variance accumulated from steps 20 to 1000 for comparing the two sampling cases of $\alpha$ with the diagonal value of $R = 0.01$. The variances of both (a) state $Var(x)$ and (b) parameter $Var(\theta)$ are averaged over the dimension, that is, $N_x = 1000$ and $N_\theta = 1$, for the variables and parameter, respectively, and the number of particles $N$ for each dimension, as follows:

$$\begin{aligned} \overline{Var(x^n)} &= \frac{1}{N_x}\sum_{j=1}^{N_x}\frac{1}{N}\sum_{i=1}^{N}\left(x_i^n - \overline{x^n}\right)_j^2, \\ \overline{Var(\theta^n)} &= \frac{1}{N}\sum_{i=1}^{N}\left(\theta_i^n - \overline{\theta^n}\right)^2, \end{aligned} \tag{3.56}$$

where the index $j$ denotes the elements of state $x$ and $\overline{x^n}$ and $\overline{\theta^n}$ are the ensemble

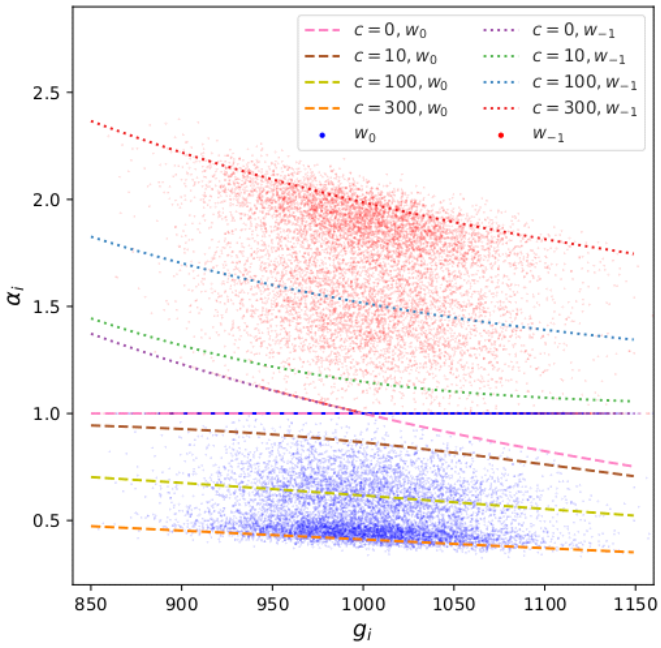


Figure 3.2: Examples of the calculated $\alpha_i$ and the analytical solutions using the Lambert W function. Calculated values represent an accumulation of 1000th steps, divided into $W_0$ branch (blue points) and $W_{-1}$ branch (red points). The analytical solutions are shown as a function of $g_i$ with different log-weight offsets $c_i$ = 0, 10, 100, and 300.

means. Note that the dimension of parameter $\theta$ is one. The true variances based on the solution of Eq. (3.55) are shown as "True." From these comparisons, both the state and parameter variances are close to the "True" value when sampling 50% from the $\alpha_i \leq 1$ branch. In contrast, when sampling from only the $\alpha \leq 1$ branch and only the $\alpha \geq 1$ branch, we see that the variance becomes smaller and larger, respectively, with the same trend as for [ZvLA16].

Figure 3.4 compares the posterior pdf obtained in the 50% sampling case with the true pdf for the diagonal value of $R$ = 0.01. Because the ensemble size is too small compared to the number of model dimensions, both of the estimated pdfs are shown as histograms accumulated over the time evolution from steps 20 to 1000 for the state and parameter. From Figure 3.4 (a) and (b), we see that the obtained pdfs for state $x_1$ and parameter $\theta$ are close to the true pdf.

These results indicate that the method of combining the parameter vector with the

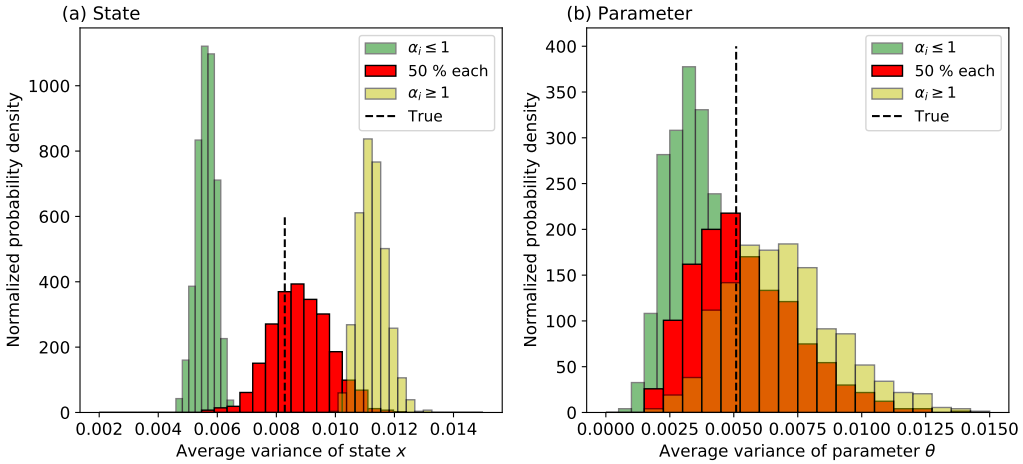


Figure 3.3: Histograms of cumulative variance comparing diagonal values of $R = 0.01$ for (a) state and (b) parameter, respectively. Three sampling percentages from the $\alpha \leq 1$ branch – 100%, 50%, and 0% – are compared with the true variance (dashed line).

state vector of IEWPF using an augmented state space model is effective. Then, the variance and shape of the posterior pdf for the parameter are also close to those of the true pdf under the condition that the variance and shape of the posterior pdf for the state are close to those of the true pdf.

### 3.3.2 Lorenz 96 model with parameterized forcing

The Lorenz 96 model [Lor96] has a chaotic (i.e., nonlinear) behavior that mimics weather phenomena with advection, dissipation, and forcing terms. It is widely used for testing and benchmarking data assimilation algorithms as a toy model because of its dimensional extensibility and ease of implementation. From this perspective, this study uses the Lorenz 96 model with a parameterized forcing term as the time evolution model expressed in Eq. (3.1) to explore the validity of the IEWPF method with nudging in a nonlinear high-dimensional system. The model equation is given by

$$\frac{d}{dt}x_j = (x_{j+1} - x_{j-2})x_{j-1} - x_j + F_j, \tag{3.57}$$

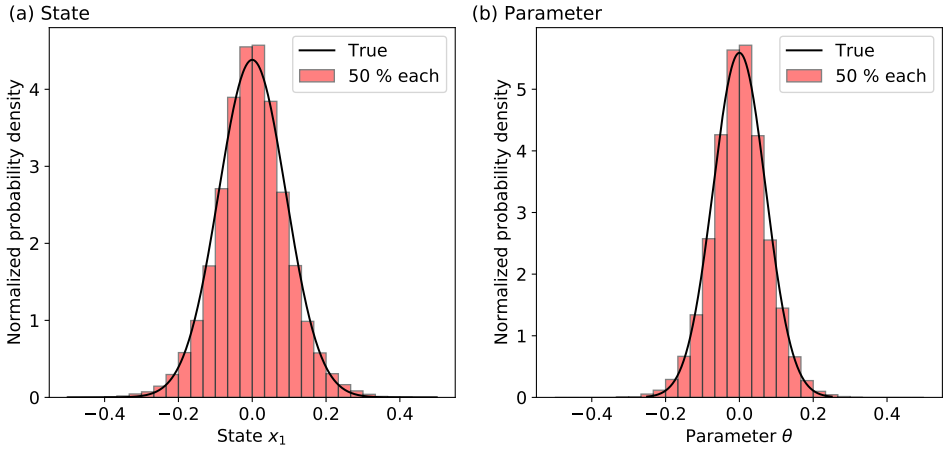


Figure 3.4: Posterior pdfs represented by particles using 50% sampling case compared with true pdf (full line) for (a) state $x_1$ of element one and (b) parameter $\theta$.

where index $j = 1, \ldots, N_x$ with cyclic indices, $x_j$ is the state variable for the model at position $j$, $N_x$ is total dimension, and $F_j$ is the forcing term parameterized by

$$F_j(\theta_0, \theta_1, \theta_2) = c_0\theta_0 + c_1\theta_1 \sin\left(\frac{2\pi}{c_2\theta_2} j\right), \tag{3.58}$$

for which $c_0, c_1$, and $c_2$ are true values and $\theta_0, \theta_1$, and $\theta_2$ are their respective scale parameters that have to be estimated. For the evaluation of nonlinearity, the value of $F_j$, which is typically chosen to be 8 or more to generate chaotic behavior, is set as follows. The values of $c_0$ and $c_1$ are set to 8 and 4 respectively, and $c_2$ is set to the same value as the dimension of the model state: $N_x$. Then, the scale parameters $\theta_0, \theta_1$, and $\theta_2$ are estimated, and their true values are found to be 1 each. By introducing this parameterized forcing term $F_j(\theta_0, \theta_1, \theta_2)$, each state variable $x_j$ exhibits parameter-dependent chaotic behavior. This model is numerically solved by the fourth-order Runge-Kutta scheme with a time step of $\Delta_t = 0.05$.

The procedure for the following experiment is the same as for the previous linear model. The true model error covariance matrix $Q_\beta$ for states is chosen as a tridiagonal matrix, the main diagonal value being 0.10 and both sub- and super-diagonal values being 0.025. The background error covariance matrix for the states $B_x$ and the parameter

Table 3.1: Summary of conventional and proposed methods shown in Section 3.2.

| MH1 | MH2 | MH3 |
|---|---|---|
| Conventional augmented method expressed as Eq. (3.2) | Proposed state-space model expressed as Eq. (3.5) with the covariance matrix $\tilde{Q}$ | MH2 and Adam method-based nudging described in Section 3.2.5 |

$B_\theta$ are chosen as a diagonal matrix with the main diagonal values 1 and 0, respectively. In the experiments below, the true observation error matrix $R$ is diagonal, with main diagonal values of 0.02. For the assimilation, we choose the same matrices $Q_\beta$, $B_x$, and $R$ as when the observation was generated, namely, the true one. The matrices $Q_\eta$ and $B_\theta$ for parameters are diagonal matrices with main values of $5.0 \times 10^{-6}$ and 0.001, respectively. The step size $\lambda$ for the Adam method is set to 0.001. As in the linear model, the number of particles is set to $N = 20$. To consider high-dimensional cases, $N_x$ is chosen as 1000, as in the linear model experiment.

In contrast to the previous evaluation using the linear model and a static parameter, this experiment investigates the ability of the present method to estimate time-varying (i.e., dynamic) parameters in nonlinear high-dimensional systems. In the experiment, all model states are observed every fourth step (i.e., the assimilation interval is 4). Moreover, this 1000-dimensional evaluation with only 20 particles can validate its applicability to realistic geophysical, climate, and other problems. First, we compare the methods outlined in Section 3.2 in terms of the root mean square error (RMSE) and the ensemble spread. Table 3.1 summarizes the three comparison methods. Next, we compare the impact of the parameter error covariance $Q_\eta$ and the step-size factor $\lambda$ on the ensemble. The performance indicator for parameter estimation is not only the RMSE but also the ratio of the RMSE to the spread in the ensemble, and it is preferable that their ratio become one for Gaussian variables. This indicates that the RMSE, which represents the forecast error, and the spread, which represents the uncertainty of the forecast, are in agreement, i.e., the forecast is neither under- nor overestimated, which is appropriate. Note that for non-Gaussian variables, this is only true for the forecast ensemble [FAAT14].

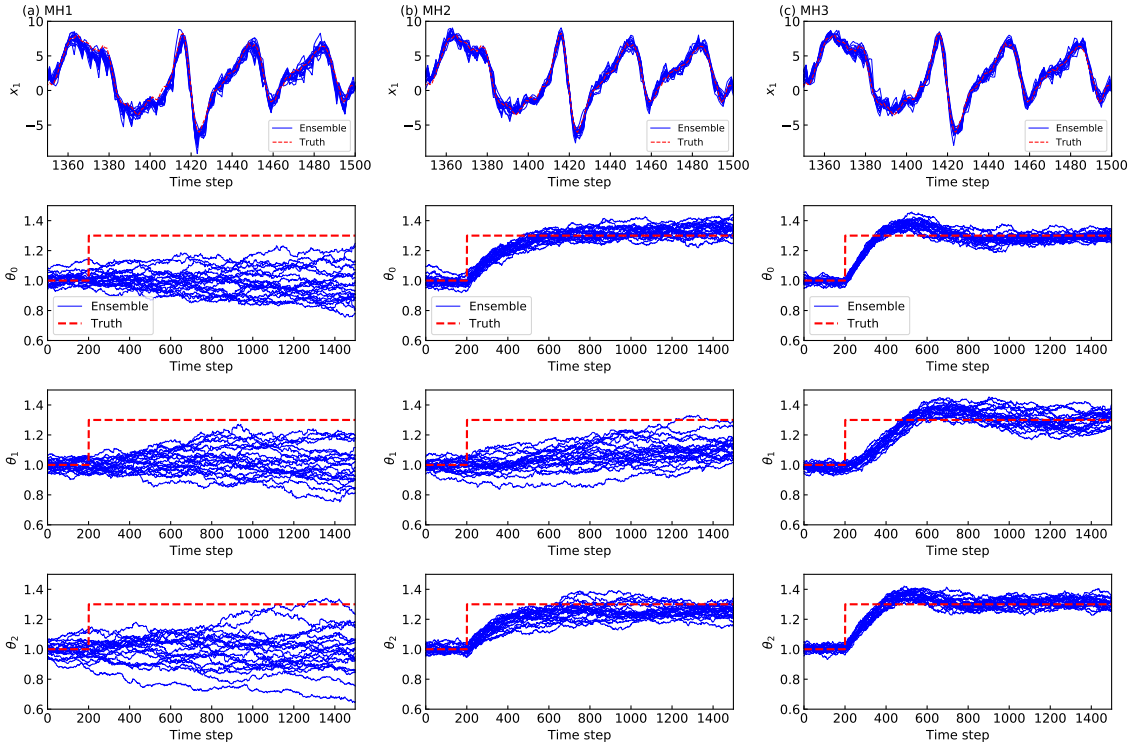


Figure 3.5: Comparison of estimated state and parameter trajectories between (a) the conventional augmented method (MH1), (b) the new method without nudging ($\lambda = 0$) (MH2), and (c) the new method with nudging ($\lambda = 0.001$) (MH3). The solid lines (blue) show each of the 20 ensemble members, and the dashed lines (red) show the true parameter value. Only steps 1350–1500 are shown for the state, and each true parameter is increased by 30% at step 200.

### Comparison of the methods

Figure 3.5 compares the true values and particle trajectories for the three methods described above for the state $x_1$ and the three scale parameters $\theta_0$, $\theta_1$, and $\theta_2$. All variables are observed every four steps, setting the main diagonal value of matrix $R$ to 0.02. Each true parameter is increased by 30% at step 200, as the dashed red line shows. The figure shows the difference in tracking performance of the three methods for abrupt parameter changes and the advantage of the present method. Method MH1 shown in Figure 3.5 (a) is the conventional augmented method expressed as Eq. (3.2). There are some steps where the trajectories of each ensemble deviate from the true trajectory in the state, and the ensemble spreads out greatly and cannot track abrupt changes in

all three parameters. Then, both methods MH2 and MH3 shown in Figure 3.5 (b) and (c), respectively, are based on the proposed state-space model expressed as Eq. (3.5) with the covariance matrix $\tilde{Q}$. Method MH3 shown in Figure 3.5 (c) further applies the Adam method-based nudging described in Section 3.2.5 with step-size factor $\lambda = 0.001$. The results for the state show that the trajectories of each ensemble are close to the true trajectory. Although both methods tend to approach the true values for $\theta_0$ and $\theta_2$, the Adam method-based nudging is more accurate and responsive to abrupt changes, especially for $\theta_1$.

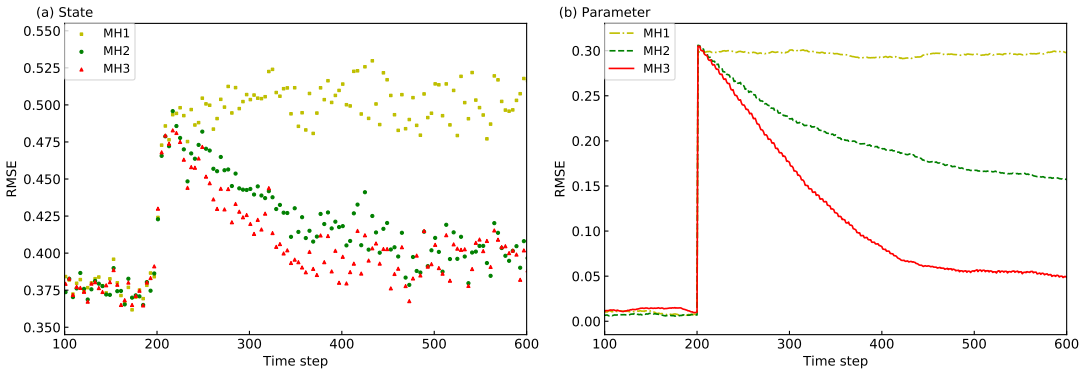


Figure 3.6: Comparison of time series RMSE after an abrupt parameter change (step 200 step) between augmented method MH1, method MH2 without nudging ($\lambda = 0$), and method MH3 with nudging ($\lambda = 0.001$), as per Figure 3.5. The third step after the filtering for the (a) state and all steps for the (b) parameter are shown. Each value is averaged over all elements.

Figure 3.6 (a) and (b) shows comparisons of the time series RMSE for the states and parameters, respectively. The horizontal axis indicates time steps 100–600, where the difference between methods is significant in Figure 3.5. For the state, because the assimilation interval is four, each value represents the average of all elements (i.e., 1000) for the third step, which has the largest prediction error after filtering. For the parameter, the average values of all elements (i.e., 3) for all steps are shown. The results show that the estimation error for the parameters after the abrupt parameter change (step 200) increases the error in the forecast step of the model states, and the estimation error of method MH3 decreases the fastest for both states and parameters. Figures 3.7(a) and (b) show the RMSE and spread comparisons for the states and parameters, respec-

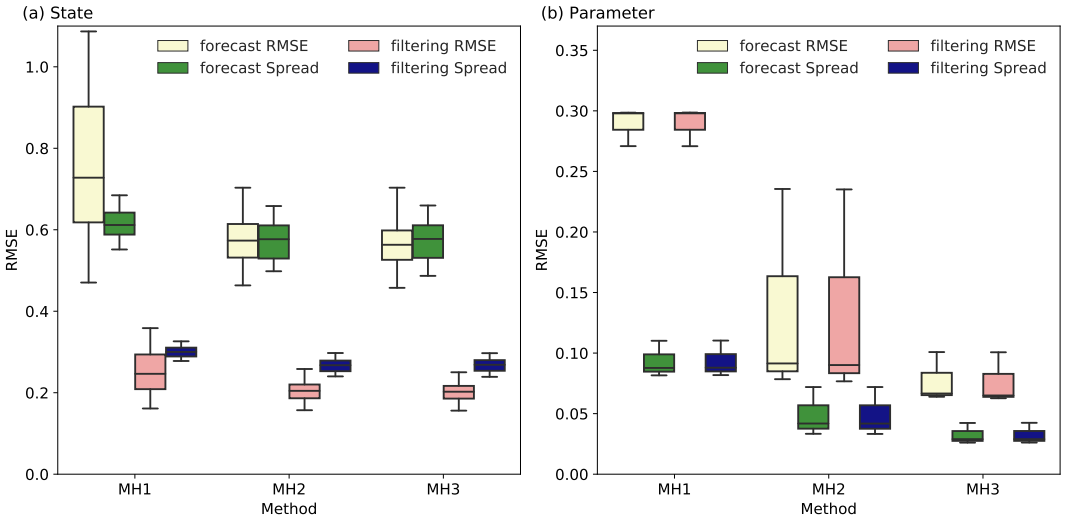


Figure 3.7: Box plots showing comparisons of RMSE and spread for forecast and filtered ensembles between augmented method MH1, method MH2 without nudging ($\lambda = 0$), and method MH3 with nudging ($\lambda = 0.001$), as per Figure 3.5. Each IQR indicates the dispersion of the (a) state and (b) parameter elements averaged over the forecast and filtering in steps 100–1500. Outliers are not plotted.

tively. Each box plot shows the time-averaged RMSE and spread at the forecast and filtering steps 100–1500 shown in Figure 3.5, including the abrupt change (at step 200). Therefore, the interquartile range (IQR) of the box plot indicates the dispersion across the dimensions of the model states (1000) and parameters (3). Note that outliers are not plotted to exclude estimation errors immediately after the abrupt changes at step 200. From the result for the states shown in Figure 3.7 (a), the new methods (i.e., MH2 and MH3) have smaller RMSE values and dispersion than those of the conventional method (i.e., MH1), especially in the forecast step. The result for the parameters shown in Figure 3.7 (b) clearly shows that both the RMSE values and dispersion of MH3 (i.e., with nudging) are smaller than the others, and the spread is also smaller. The fact that the RMSE dispersion for MH3 is smaller than that for MH2 means that the difference in RMSE in the three parameters is small. Thus, the Adam-based nudging method reduces differences in estimation accuracy for each parameter, which shows the effectiveness of combining IEWPF with Adam.

Table 3.2: Summary of experimental conditions

| Conditions | exp1 | exp2 | exp3 | exp4 |
|---|---|---|---|---|
| $\sigma_\eta^2$ | $1.0 \times 10^{-6}$ | $5.0 \times 10^{-6}$ | $1.0 \times 10^{-5}$ | $5.0 \times 10^{-5}$ |
| $\lambda$ | 0.001 | 0.001 | 0.001 | 0.001 |
| $R$ | 0.02 | 0.02 | 0.02 | 0.02 |
| Observation | all | all | all | all |

| Conditions | exp5 | exp6 | exp7 | exp8 | exp9 |
|---|---|---|---|---|---|
| $\sigma_\eta^2$ | $5.0 \times 10^{-6}$ | $5.0 \times 10^{-6}$ | $5.0 \times 10^{-6}$ | $5.0 \times 10^{-6}$ | $5.0 \times 10^{-6}$ |
| $\lambda$ | 0.0005 | 0.002 | 0.004 | 0.004 | 0.004 |
| $R$ | 0.02 | 0.02 | 0.02 | 0.08 | 0.02 |
| Observation | all | all | all | all | half |

### Dependence of parameter error covariance and step-size factor

In the following, we investigate the impact of the parameter error covariance $Q_\eta$ and the step-size factor $\lambda$ on the estimation accuracy (RMSE) and the ensemble spread. Table 3.2 summarizes the conditions of the subsequent experiments.

Figure 3.8 and Figure 3.10 show the true values and the particle trajectories for the scale parameter $\theta_0$ under the combination of different values of $Q_\eta$ and $\lambda$, respectively. Note that $Q_\eta$ is chosen as a diagonal matrix and is denoted as $Q_\eta = \sigma_\eta^2 I$. The graph

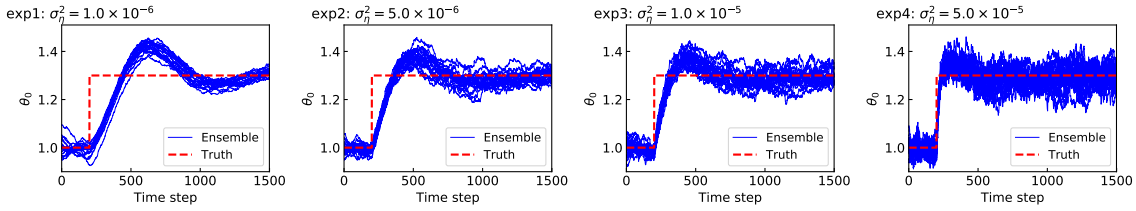


Figure 3.8: Comparison of estimated parameter trajectories between different values of $\sigma_\eta^2$: $1.0 \times 10^{-6}$ (exp1), $5.0 \times 10^{-6}$ (exp2), $1.0 \times 10^{-5}$ (exp3), and $5.0 \times 10^{-5}$ (exp4) under the same value of $\lambda = 0.001$. The solid lines (blue) show each of the 20 ensemble members, and the dashed lines (red) show the true parameter value. Each true parameter is increased by 30% at step 200.

labeled exp2 in Figure 3.8 is the reference condition with $\sigma_\eta^2 = 5.0 \times 10^{-6}$ and $\lambda = 0.001$; it is the same graph shown in Figure 3.5 (c) $\theta_0$. The other graphs exp1, exp3, and

exp4 in Figure 3.8 show the cases where $\sigma_\eta^2$ is $1.0 \times 10^{-6}$, $1.0 \times 10^{-5}$, and $5.0 \times 10^{-5}$, respectively, under the same value of $\lambda = 0.001$. These graphs show that the larger the parameter covariance, the larger the ensemble spread and the less overshoot after the abrupt parameter change.

Next, we quantitatively evaluate the impact of the parameter error covariance $Q_\eta$ on the ensemble. Figure 3.9 shows the dependence of the parameter error covariance $Q_\eta$ on the RMSE and spread for (a) states and (b) parameters. Each box plot shows the time-averaged RMSE and spread at the forecast and filtering steps 100–1500. The forecast RMSE and spread include three cycles of forecast steps because the filtering interval is four. The four values of $\sigma_\eta^2$ shown on the horizontal axis are for exp1, exp2, exp3, and exp4 in Figure 3.8. Note that outliers are not plotted to exclude estimation errors immediately after abrupt changes in step 200. For the states, we can see from

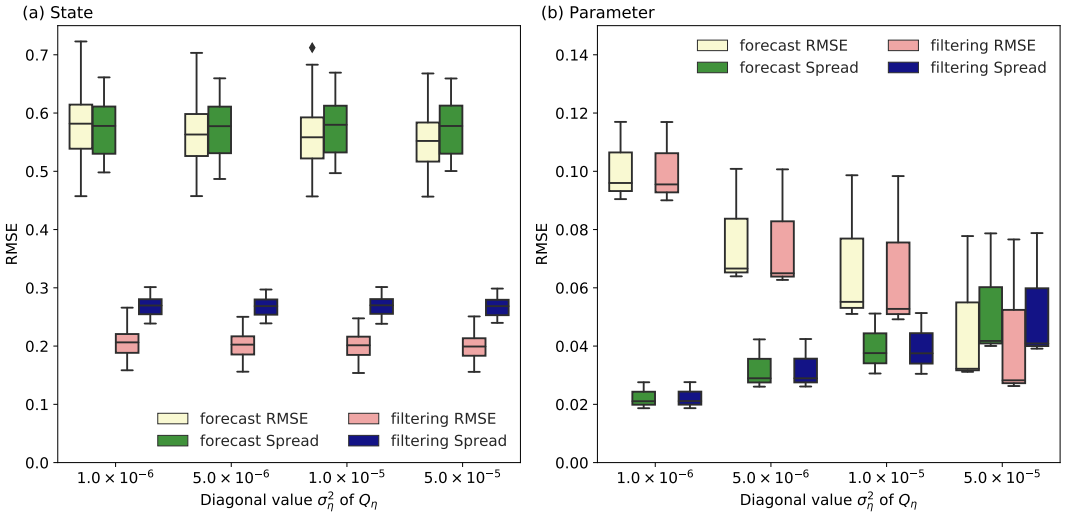


Figure 3.9: Box plots showing comparison of RMSE and spread for each of forecast and filtered ensembles between different values of $\sigma_\eta^2 = 1.0 \times 10^{-6}, 5.0 \times 10^{-6}, 1.0 \times 10^{-5}$, and $5.0 \times 10^{-5}$, as for Figure 3.8. Each IQR indicates the dispersion of the (a) state and (b) parameter elements averaged over the forecast and filtering steps 100–1500. Outliers are not plotted.

Figure 3.9 (a) that neither the RMSE value nor the spread value depends on the diagonal value for the parameter error covariance $Q_\eta$. In addition, the values of the forecast RMSE and spread are close, that is, their ratio is close to one. In contrast, for the

parameters, Figure 3.9 (b) shows that as the diagonal values $\sigma_\eta^2$ increase, the spread values also increase, and the RMSE values decrease. Especially in the case of $\sigma_\eta^2 = 5.0 \times 10^{-5}$, the values of the forecast RMSE and spread are close, that is, their ratio is close to one.

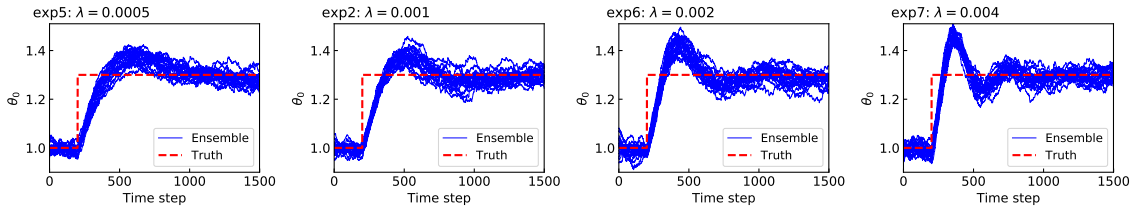


Figure 3.10: Comparison of estimated parameter trajectories between different values of $\lambda$: 0.0005 (exp5), 0.001 (exp2), 0.002 (exp6), and 0.004 (exp7) under same value of $\sigma_\eta^2 = 5.0 \times 10^{-6}$. The solid lines (blue) show each of the 20 ensemble members, and the dashed lines (red) show the true parameter value. Each true parameter is increased by 30% at step 200.

The graph shown in Figure 3.10 exp2 is the same graph in Figure 3.8 exp2 for the reference condition with $\sigma_\eta^2 = 5.0 \times 10^{-6}$ and $\lambda = 0.001$. In cases exp5, exp6, and exp7 in Figure 3.10, $\lambda$ = 0.0005, 0.002, and 0.004, respectively, under the same value of $\sigma_\eta^2 = 5.0 \times 10^{-6}$. These graphs show that the larger the step-size factor, the faster the value approaches the true value after the abrupt change, but the more likely it is to overshoot.

Figure 3.11 shows the dependence of the step-size factor $\lambda$ on the RMSE and spread for (a) states and (b) parameters. Each box plot shows the time-averaged RMSE and spread at the forecast and filtering steps 100–1500, and the forecast RMSE and spread include three cycles of forecast steps, as in Figure 3.9. The four values of $\lambda$ shown on the horizontal axis are for exp5, exp2, exp6, and exp7 in Figure 3.10. Note that outliers are not plotted, as in Figure 3.9. Similar to the trend shown in Figure 3.9, there is almost no dependence of the step-size factor $\lambda$ on the RMSE and spread for states. For parameters, the spread does not increase even as the step-size factor $\lambda$ increases, but the RMSE decreases, that is, the ratio of the forecast RMSE to the spread gradually approaches one.

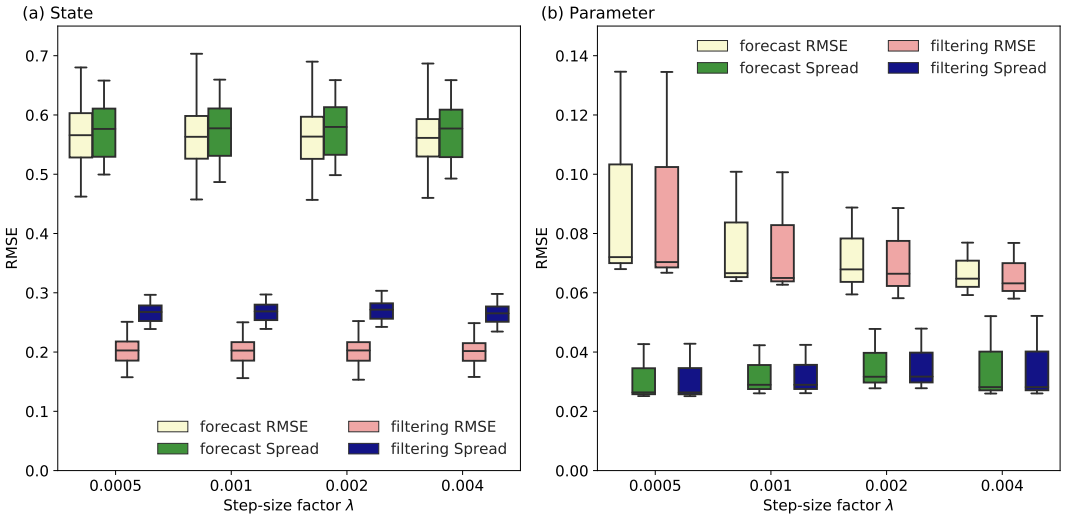


Figure 3.11: Box plots showing comparison of RMSE and spread for each of forecast and filtered ensembles between different values of $\lambda = 0.0005$, 0.001, 0.002, and 0.004, as for Figure 3.10. Each IQR indicates the dispersion of the (a) state and (b) parameter elements averaged over forecast and filtering steps 100–1500. Outliers are not plotted.

### Dependence of observation error and number of observations

To evaluate the dependence of the observation error and number of observations, we compare the large step size condition, $\lambda = 0.004$ (exp7), with two additional experiments (exp8 and exp9). In case exp8, the main diagonal value for the matrix $R$ is large, and in the following, the value is set to 0.08. Note that experiment exp8 uses observation data generated at $R = 0.08$. Hence, $R$ for data generation and assimilation are the same value. In the second case exp9, the state is observed at every other grid point, so that $h(x^n) = \left(x_1^n, x_3^n, \ldots\right)^T$. In both the additional experiments, the step size and the diagonal value for the parameter error covariance are the same as for exp7: $\lambda = 0.004$ and $\sigma_\eta^2 = 5.0 \times 10^{-6}$.

Figure 3.12 shows a comparison of the RMSE and spread for different observation conditions for the (a) state and (b) parameter. The description of the box plot is the same as in Figure 3.11. In Figure 3.12, exp7 shows the results of the reference condition, $R = 0.02$, and all model states are observed. From the comparison of the exp7 and exp8 cases in Figure 3.12 (a), the change in $R$ from 0.02 to 0.08 increases both the RMSE and spread,

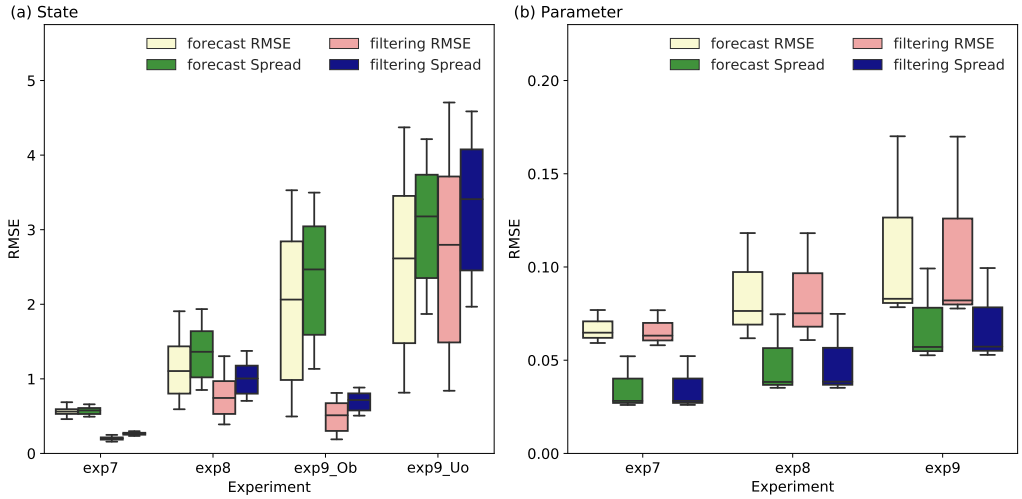


Figure 3.12: Box plots showing comparisons of the RMSE and spread for forecast and filtered ensembles between the large step size condition (exp7) and two large observation error cases: $R = 0.08$ (exp8) and partially observed (exp9). Each IQR indicates the dispersion of the (a) state and (b) parameter elements averaged over forecast and filtering steps 100–1500. Outliers are not plotted. "Ob" and "Uo" represent observed and unobserved states, respectively.

but the spread increase is somewhat more pronounced. For the parameter in Figure 3.12 (b), RMSE values and dispersion tend to increase compared to the spread. From comparison of cases exp7 and exp9 in Figure 3.12 (a), because the number of observed variables was reduced to half, both the RMSE and spread are increasing except for the filtering value of the observed variable. As for the parameters, both the RMSE and spread show a small increase in their median value, but an increase in dispersion. The results indicate that increasing the observation error and decreasing the observation density increase the differences in estimation accuracy between parameters. In other words, the decrease in observed information reduces the estimation accuracy for the parameters with little impact (i.e., low sensitivity) on the model state. Since the RMSE was reduced by increasing the step size and parameter error covariance in the previous experiment, a reduction in the error difference between parameters can be expected even at this low observation density.

## 3.4 Conclusion

This chapter described a resilient and efficient state and time-varying parameter estimation method for nonlinear high-dimensional systems through a sequential data assimilation process. First, we introduced an extension of IEWPF to an augmented state-space model with a correlated covariance matrix. We then presented an IEWPF-based method that incorporates a nudging technique inspired by optimization algorithms in machine learning into the parameter time evolution model by using the flexibility of the proposal density in particle filtering.

The performance of the method is examined in a 1000-dimensional linear model and the nonlinear Lorenz 96 model. Experiments on the linear model with the static parameter indicate that the impact of the scalar factor $\alpha$ on the variance of the parameter is similar to that on the variance of the state. Numerically, under the condition that the variance and shape of the posterior pdf for the states are close to the true ones, those for the parameter are also close to the true ones.

The experimental results for the nonlinear Lorenz 96 model with time-varying parameters show the following points. First, the presented state augmentation method successfully estimates states and parameters simultaneously, even when the number of ensemble members is much smaller than the model dimension. This result indicates that filter degeneracy is avoided when extended to an augmented state-space model. Second, the parameter nudging method inspired by optimization algorithms accelerates the tracking of abrupt parameter changes and reduces the difference in estimation accuracy for each parameter. This result suggests the effectiveness of combining the IEWPF with the Adam optimization algorithm. Third, from evaluating the impact of the parameter error covariance and the step-size factor on the time-averaged RMSE and the ensemble spread, the former increases the spread and decreases the RMSE, while the latter decreases the RMSE. Properly determining these values so that the ratio of the RMSE to the spread approaches one will allow for good ensemble generation. However, a systematic method for this determination is a subject of future research. Finally, from evaluating the dependence of the observation error and the number of observations, the decrease in observed information reduced the estimation accuracy for parameters with little impact (i.e., low sensitivity) on the model state. This could potentially be mitigated by adjusting the step-size factor and the parameter error co-

variance. Alternatively, it may be beneficial to narrow the parameters to be estimated to those with high sensitivity through a preliminary sensitivity analysis.

In the numerical experiments in this chapter, the Lorenz 96 model with parameterized forcing was used mainly to evaluate the nonlinearity of the time evolution of the model states, but further investigation of the nonlinearity of the parameters is needed. Adam optimization is a first-order gradient-based method, and it is widely used to learn the weights in deep neural networks, that is, nonlinear functions. Thus, our Adam-based nudging term can work theoretically in nonlinear problems. However, even for nonlinear convex problems, there are conditions and limits to convergence, and new methods have been proposed [RKK19]. Furthermore, convergence for non-convex problems is still an open question, though [CLSH18] developed an analysis framework and a set of sufficient conditions that guarantee convergence. Therefore, the applicability of the IEWPF method with nudging to various nonlinear problems in data assimilation needs to be investigated. In particular, future research topics include how well the first-order approximation of the parameters introduced in Eq. (3.3) and the gradient method used for parameter nudging can handle the nonlinearity associated with the parameters.

In this chapter, we applied the new online parameter estimation scheme to IEWPF as an example of a PF that can avoid filter degeneracy. The method is shown to be capable of resilient and efficient parameter estimation for time-varying parameters. The results lead to the conjecture that the new method is applicable to realistic geophysical, climate, and other problems. Because several approaches have been proposed to avoid filter degeneracy (e.g., [SEvLA19]), the evaluation of another combination is a subject of future research.

# 4

# Posterior distribution estimation in implicit equal-weights particle filter

The previous chapters described an extension of the implicit equal-weights particle filter (IEWPF) to parameter estimation. This chapter shows how to estimate inherent coefficients (factors) in IEWPF from the perspective of eliminating the need for prior assumptions and tuning. We propose a method for adaptively determining the factor by iterative computation using analytical solutions of the Lambert W function and the Kullback-Leibler (KL) divergence.

## 4.1 Introduction

A simple way to solve the filter degeneracy problem is to ensure that all particle weights are equal. IEWPF, which sets the target weights to the minimum of the optimal proposal weights of all particles, was proposed by [ZvLA16]. An extension of this method to parameter estimation was also presented in Chapter 3. In these methods,

each particle is sampled from a mode of optimal proposal density and scaled random perturbations. The scalar factor here is chosen so that the weight of each particle equals the target weight. In other words, the factor must be determined so that each particle reaches the target weight. There are two issues in determining this factor. First, this structure is known to exclude part of the state space for all particles except one (i.e., the particle with the target weight) [vLKN+19]. Specifically, because the size of the random vector is found deterministically, the proposal density misses one degree of freedom for all particles except one, which is the smallest weighted particle. In a typical high-dimensional system, the effect of missing one degree of freedom can usually be tolerated but leads to bias.

The second issue is the computational difficulty of determining the factor to satisfy the equation that makes the weights of all particles equal to the target weight. This equation can be solved iteratively, but computation time can be an issue. Therefore, [ZvLA16] used analytical solutions based on the so-called Lambert W function [CGH+96]. Note that because the Lambert W function is a multivalued function, the solution must be uniquely determined. In the methods described in [ZvLA16] and Chapter 3, 50% of each is selected from the two branches based on the results of comparison with the true probability density function (pdf) in the linear model. However, its validity has not been theoretically proven or demonstrated, especially for nonlinear models and parameter estimation applications. As an alternative approach, a revised IEWPF was proposed [SEvLA19], in which only one branch of the Lambert W function is adopted as a solution of the factor. However, there are additional parameters that need to be adjusted appropriately.

In light of the above, it is challenging to apply IEWPF to cases where prior assumptions or prior adjustment of values using cases with known true values is not possible, that is, for real problems. Therefore, this chapter presents posterior distribution estimation in IEWPF using KL divergence. Although the analytical solution of the Lambert W function is used, no prior assumptions are made as to which branch to select (e.g., 50% selection), and the decision is made adaptively through iterative calculation. This chapter makes three main contributions to distribution estimation in IEWPF:

- We describe a sequential estimation method for the scalar factor $\alpha$ that determines the variance of the posterior distribution in IEWPF.

- Converting the particle distribution into a histogram makes it possible to minimize the KL distance with a low computational load.

- Evaluation using the linear model confirmed that the variance value for the estimated posterior distribution was almost the same as the analytical value. Also, evaluation using the nonlinear Lorenz 96 model confirmed that the same or better accuracy and ensemble quality (i.e., the ratio of RMSE and spread) could be obtained without making any prior assumptions about the scalar factor $\alpha$.

## 4.2 Preliminaries

### 4.2.1 Proposal density particle filter

This section discusses the need to introduce the proposal density and its methodology. A conventional PF draws particles from the prior distribution. These are then modified to be particles of the posterior distribution by weighting based on likelihood. The extreme increase in the weights of certain particles compared to the weights of others (i.e., filter degeneracy), described in the introduction, is due to a discrepancy between the prior and posterior distributions. The idea of the proposal density particle filter is to freely change the model equations to move the particles of the prior distribution to the necessary part of the state space. This is possible because the numerical model is stochastic rather than deterministic. For a detailed explanation of the proposal density particle filter, see Section 2.1.3. We assume that $x^n \in \mathbb{R}^{N_x}$ are model states and $p\left(x^n|x^{n-1}\right)$ is the transition density, which is the pdf for the state at time $n$ when the state at time $n-1$ is given. When the proposal density $q\left(x^n|x_i^{n-1}, y^n\right)$ is introduced, particles at time step $n$ are allowed to arise according to different models as expressed in Eq. (2.12). Then, the posterior distribution can be rewritten as in Eq. (2.13). Using the particle expression in Eq. (2.6), the posterior distribution is represented as in Eq. (2.16).

If the optimal proposal distribution in Eq. (2.19) is chosen as the specific design of the proposal distribution $q\left(x^n|x_i^{n-1}, y^n\right)$, the variance of the weights is minimized [SBM15]. We further assume a Gaussian model; model and observation error with mean zero and covariance $Q$ and $R$, respectively; and the linear observation operator

$\tilde{H}$. This allows us to express the optimal proposal density as

$$p(x^n|x_i^{n-1}, y^n) = \mathcal{N}\left(\zeta_i^n, P\right), \tag{4.1}$$

where

$$\begin{aligned} \zeta_i^n &= f(x_i^{n-1}) + Q\tilde{H}^T\left(\tilde{H}Q\tilde{H}^T + R\right)^{-1}\left\{y^n - \tilde{H}f\left(x_i^{n-1}\right)\right\}, \\ P &= \left(I - Q\tilde{H}^T\left(\tilde{H}Q\tilde{H}^T + R\right)^{-1}\tilde{H}\right)Q. \end{aligned} \tag{4.2}$$

Then, the particle $i$ sampled from the optimal proposal density is obtained by

$$x_i^n = \zeta_i^n + P^{1/2}\xi_i^n\ , \quad \xi_i^n \sim \mathcal{N}(0, I). \tag{4.3}$$

### 4.2.2 IEWPF

In IEWPF proposed by [ZvLA16], the target weights are set equal to the minimum of the optimal proposal weights for all particles. Each particle is set to the mode of optimal proposal density plus a scaled random perturbation, expressed as

$$x_i^n = \zeta_i^n + \alpha_i^{1/2}P^{1/2}\xi_i^n, \tag{4.4}$$

where $\zeta_i^n$ represents the mode of $q\left(x^n|x_i^{n-1}, y^n\right)$, $P$ is a measure of the width of that pdf, and $\alpha_i$ is a particle-specific scalar factor chosen so that the weight of each particle equals the target weight $w_{target}$. Specifically, the factor $\alpha_i$ determined to satisfy Eq. (2.42). Note that the only formal difference here compared to Eq. (4.3) is the factor $\alpha_i$.

Under the assumption of a linear observation model and Gaussian error, the equation that the factor $\alpha_i$ of the particle $i$ must satisfy is as follows:

$$(\alpha_i - 1)\,\xi_i^{nT}\xi_i^n - N_x \log \alpha_i - 2\log\left(\left|1 + \frac{\partial \alpha_i^{1/2}}{\partial \xi_i^n}\frac{\xi_i^n}{\alpha_i^{1/2}}\right|\right) = C - \phi_i, \tag{4.5}$$

where $C$ denotes a constant term and $\phi_i$ is obtained from Eq. (2.47). The right side of Eq. (4.5) expresses the log-weight offsets from the target weight for each particle $i$ and can be obtained practically from $c_i = \max_j\{\phi_j\} - \phi_i$. For the weights based on the different likelihoods of each particle to reach the target weights, the weight offset $c_i$ ($i = 1, \ldots, N$) requires $c_i \geq 0$. Because solving the equations numerically for each particle

is inefficient, each $\alpha_i$ is obtained by an analytical solution of the Lambert W function under a high-dimensional approximation. For a detailed explanation, see Section 2.1.6.

## 4.3 Methodology

### 4.3.1 Problem statement

To evaluate state and time-varying parameter estimations in nonlinear high-dimensional systems, we consider the same state-space model as in Chapter 3. The system model is expressed as follows:

$$\begin{aligned} z^n \equiv \begin{pmatrix} x^n \\ \theta^{n-1} \end{pmatrix} &= \begin{pmatrix} f\left(x^{n-1}, \theta^{n-2}\right) \\ \theta^{n-2} \end{pmatrix} + \begin{pmatrix} \frac{\partial f}{\partial \theta}\Big|_{n-2} \eta^{n-1} + \beta^n \\ \eta^{n-1} \end{pmatrix} \\ &\equiv \tilde{f}\left(z^{n-1}\right) + \tilde{\rho}^n, \end{aligned} \tag{4.6}$$

where the notation of the augmented vector $z^n = \left[x^{n^T}, \theta^{n-1^T}\right]^T$, model $\tilde{f}$, and perturbation $\tilde{\rho}$ are introduced. Here, the augmented perturbation $\tilde{\rho}$ can be drawn from the error pdf $N\left(0, \tilde{Q}_n\right)$ expressed as in Eq. (3.11). When the linear observation model $\tilde{H}$ is assumed, Eq. (4.4) can be rewritten as

$$z_i^n = \zeta_i^n + \alpha_i^{1/2} P^{1/2} \xi_i^n, \tag{4.7}$$

where $\zeta_i^n$ and $P$ are obtained from the optimal proposal density shown in Eq. (4.1) as follows:

$$\begin{aligned} \zeta_i^n &= \tilde{f}\left(z_i^{n-1}\right) + \tilde{Q}\tilde{H}^T\left(\tilde{H}\tilde{Q}\tilde{H}^T + R\right)^{-1}\left\{y^n - \tilde{H}\tilde{f}\left(z_i^{n-1}\right)\right\}, \\ P &= \left(\tilde{Q}^{-1} + \tilde{H}^T R^{-1}\tilde{H}\right)^{-1}. \end{aligned} \tag{4.8}$$

The objective is to optimally determine the scalar factor $\alpha_i$ for each particle. The Lambert W function gives an analytical solution under a high-dimensional approximation, which has two branches: $\alpha \geq 1$ solutions that give a large ensemble spread and $\alpha < 1$ solutions that give the opposite small spread. For each particle at each time step, it is necessary to choose which solution to employ.

### 4.3.2 $\alpha$ estimation with a reverse KL

This section explains an $\alpha$ estimation method by minimizing a reverse KL (RKL) divergence. First, because choosing a solution for $\alpha$ corresponds to determining the posterior distribution based on Eq. (4.7), we define a distribution $q(z^n|\alpha)$ with $\alpha$ as the hyperparameter. However, because the desired distribution of $q(z^n|\alpha)$ cannot be known in advance, estimating $\alpha$ based on some metric is difficult, especially if the model $f$ is nonlinear. Thus, when it is difficult to evaluate the posterior function directly, approximate inference methods such as variational inference [BKM17] are effective. The advantage of using variational inference is that the inference problem can be reformulated as an optimization problem, and various optimization techniques can be applied. In variational inference, we typically define the distribution $q$ to be estimated and minimize the RKL divergence from the desired distribution $p$, that is, $KL(q||p)$.

Next, the target distribution $\hat{p}(z^n)$ is obtained as follows. As discussed for Eq. (4.5), the log-weight offsets $c_i \geq 0$ are required for all particles to reach the target weight. Here, $c_i = 0$ when $\phi_i = \max_j\{\phi_j\}$, in other words, when the weight is at its minimum. When $c_i = 0$, $\alpha_i = 1$ is obtained from the Lambert W function that gives the approximate solution of Eq. (4.5), which represents the optimal proposal density expressed in Eq. (4.1). In the IEWPF concept, IEWPF is adjusted by an offset from the minimum weight so that all particles have equal weight. The distribution then approaches $\alpha_i = 1$ distribution, or the optimal proposal distribution, when the log-weight offset $c_i$ of all particles approaches zero. In other words, the following relationship holds:

$$\forall i \leq N, c_i \to 0 \;\Rightarrow\; \forall i \leq N, \alpha_i \to 1. \tag{4.9}$$

Therefore, we assume that the distribution when $\alpha_i = 1$ in Eq. (4.7) is the target distribution $\hat{p}(z^n)$.

Figure 4.1 shows an example of 20 particle positions representing the forecast and posterior for the linear model case in Section 4.4.1. Also shown are the observation $y^n$, the position $\zeta_i^n + \alpha_i^{1/2} P^{1/2} \xi_i^n$ with $\alpha_i$ obtained from the analytical solution expressed as Eq. (2.55), and the particles with $\alpha_i = 1$, i.e., $\zeta_i^n + P^{1/2}\xi_i^n$. Figure 4.1 (a) shows particles sampled from the forecast pdf, i.e., $\tilde{f}(z_i^{n-1}) + \tilde{\rho}_i^n$. Figure 4.1 (b) shows particles with equivalent weights sampled from the posterior distribution, i.e., $\zeta_i^n + \alpha_i^{1/2} P^{1/2} \xi_i^n$.

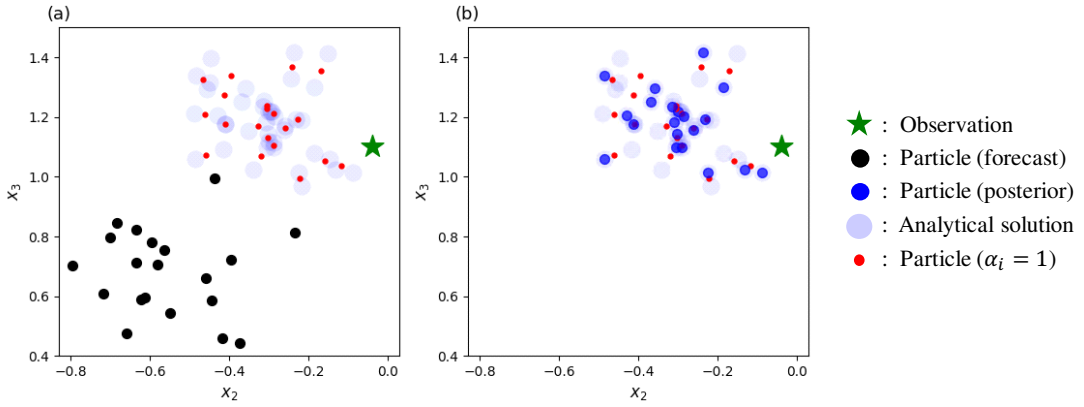


Figure 4.1: Example of 20 particle positions in the case of the linear model in Section 4.4.1:(a) particles representing the forecast $\tilde{f}(z_i^{n-1}) + \tilde{\rho}_i^n$ (black dots), (b) particles with equivalent weights representing posterior $\zeta_i^n + \alpha_i^{1/2} P^{1/2} \xi_i^n$ (blue dots), with the observation $y^n$ (star), the positions $\zeta_i^n + \alpha_i^{1/2} P^{1/2} \xi_i^n$ with $\alpha_i$ obtained from the analytical solution expressed as Eq. (2.55) (light blue circle), and particles with $\alpha_i = 1$ (red dots).

Because there are analytical solutions from two branches for each particle, the equivalent weights are satisfied by taking the position of one of them. It can be seen that the position of each particle is selected to be close to the particle with $\alpha_i = 1$ among the positions of the analytical solution. Note that when $\alpha_i = 1$, the position of each particle is different from the position of each analytical solution because the weight of each particle is not equal. In other words, the $\alpha_i = 1$ distribution can be considered the distribution obtained from the particles ignoring the weights.

Figure 4.2 shows a schematic representation of the relationship between the weights in the forecast and posterior particles. Each particle in the forecast distribution shown on the left side of the figure has a log-weight offset of $c_i > 0$ with respect to the particle at $c_i = 0$ i.e., $\alpha_i = 1$. Here, the range $R_{EW}$ of equivalent weights is the range from the minimum to the maximum of $\phi_i$ obtained from Eq. (3.27) for $2N$ solutions $\alpha_i$ obtained from the analytical solution of the Lambert W function. Namely,

$$\min_i\{\phi_i(\alpha_i)\} \leq R_{EW} \leq \max_i\{\phi_i(\alpha_i)\}, \ \alpha_i \in \alpha_{\geq 1}, \alpha_{<1}, \tag{4.10}$$

where, the sets of solutions for branches $\alpha \geq 1$ and $\alpha < 1$ are denoted as $\alpha_{\geq 1}$ and $\alpha_{<1}$, respectively. Each particle in the posterior distribution, shown on the right side of the

figure, is placed in the range of equivalent weights so that the log-weight offsets from the particle with $\alpha_i = 1$ are minimized.

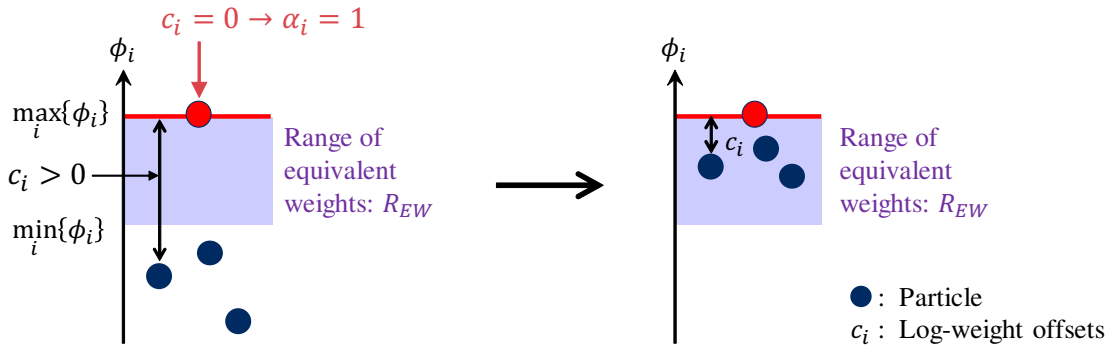


Figure 4.2: Schematic representation of the relationship between the weights of the forecast particles (left side ) and the posterior particles (right side). The two branches of the analytical solution of the Lambert W function, which is the range of equivalent weights (filled area), and the particle with the minimum weight (red circle) are shown.

Finally, we define the RKL distance to $q(z^n|\alpha)$ as $\mathrm{RKL}(\alpha) = \mathrm{KL}[q(z^n|\alpha)||\hat{p}(z^n)]$. Then, we determine $\alpha_i$ for each particle $i$ by minimizing the RKL distance at each time step as follows:

$$\underset{\alpha_i \in \alpha_{\geq 1} \text{ or } \alpha_{<1}}{\arg\min} \ \mathrm{RKL}(\alpha_i) = \underset{\alpha_i \in \alpha_{\geq 1} \text{ or } \alpha_{<1}}{\arg\min} \ \mathrm{KL}[q(z^n|\alpha_i)||\hat{p}\,(z^n)]. \tag{4.11}$$

In summary, the factor $\alpha_i$ is obtained so that the distribution in Eq. (4.7) approaches the optimal proposal distribution expressed in Eq. (4.1) while Eq. (4.5) is satisfied.

### 4.3.3 Approximation of KL calculation

In the following, we describe how to calculate the KL distance from a discrete probability distribution represented by a PF with an extremely small number of particles. Here, a discrete approximate calculation method similar to histogram density estimation is used due to the small number of particles and to reduce the computational load.

First, let $P$ and $Q$ be the probabilities obtained from the histograms corresponding to the distributions $\hat{p}\left(z_j^n\right)$ and $q\left(z_j^n|\alpha\right)$, respectively. Here, $z_j^n$ $(j = 1, \cdots, N_x)$ denotes the j-th state variable at time step $n$. Specifically, these probabilities are obtained from the value for each particle as follows. When the number of particles in the $i$-th bin $b_i$

is $n(b_i)$, the probability of the $i$-th bin is expressed as

$$P(b_i) = \frac{n(b_i) + \delta_b}{\sum_{i=1}^{N_b} [n(b_i) + \delta_b]}, \tag{4.12}$$

where $N_b$ is the number of histogram bins and $\delta_b$ is a small constant that prevents zero values ($\delta_b = 1.0 \times 10^{-6}$). In the following, the number of bins is set equal to the number of particles, that is, $N_b = N$. The histogram range is common to $p$ and $q$; the lower limit is $\min\left[p(z_j), q\left(z_j^n|\alpha\right)\right]$, and the upper limit is $\max\left[p(z_j), q\left(z_j^n|\alpha\right)\right]$ at each time step and for each variable element. The bin width is a constant value obtained by dividing this range by the number of bins.

Then, the RKL distance for step $n$ and variable $z_j$ is obtained by

$$\mathrm{RKL}_j^n(\alpha) = \sum_{i=1}^{N} Q(b_i) \log\left(\frac{Q(b_i)}{P(b_i)}\right). \tag{4.13}$$

Here, $Q(b_i)$ can be calculated in the same way as in Eq. (4.12), from the distribution of particles when $\alpha = 1$.

Finally, the value of $\mathrm{RKL}^n$ at time step $n$ adds up the $\mathrm{RKL}_j^n$ values in Eq. (4.13) across all variables.

$$\mathrm{RKL}^n(\alpha) = \sum_{j=1}^{N_x+N_\theta} \mathrm{RKL}_j^n(\alpha). \tag{4.14}$$

### 4.3.4 Iterative branch selection in original IEWPF

We adopt iterative implementation to find $\alpha$ that minimizes the RKL distance defined by Eq. (4.13). For each step $n$, a threshold $\alpha_{th}^n$ $(0 \le \alpha_{th} \le 1)$ is defined for choosing the solution from either branch $\alpha < 1$ or $\alpha \ge 1$. A uniform random number $U[0,1]$ can be used to determine $\alpha_i^n$ as follows:

$$\alpha_i^n \in \begin{cases} \alpha_{\ge 1}^n & \text{if } U[0,1] \ge \alpha_{th}^n, \\ \alpha_{<1}^n & \text{if } U[0,1] < \alpha_{th}^n. \end{cases} \tag{4.15}$$

This means that $\alpha_{th}^n$ determines the proportion of choices from each branch. Note that $\alpha_i^n$ denotes the i-th particle. For instance, when $\alpha_{th} = 0$, all samples are taken from $\alpha_{\ge 1}^n$;

when $\alpha_{th} = 1$, all samples are taken from $\alpha^n_{<1}$; and when $\alpha_{th} = 0.5$, samples are taken 50% from each branch.

Next, we explain how to determine $\alpha^n_{th}$. From the perspective of minimizing the RKL distance expressed as Eq. (4.13), $\alpha_{th}$ is updated by the following iteration:

$$\alpha^{k+1}_{th} \leftarrow \alpha^k_{th} - \lambda_\alpha \frac{\hat{m}^k}{\sqrt{\hat{v}^k} + \delta_\alpha}, \tag{4.16}$$

where $k$ is the iteration counter, $\lambda_\alpha$ is the step-size factor, and $\delta_\alpha$ is a factor that prevents division by zero ($\delta_\alpha = 1.0 \times 10^{-8}$). Here, $\hat{m}^k$ and $\hat{v}^k$ represent the moving averages of the gradient and the squared gradient, respectively, and according to [KB14], they are calculated as follows:

$$\begin{aligned} m^n &= \mu_m m^{n-1} + (1-\mu_m)\frac{\partial \mathrm{RKL}(\alpha)}{\partial \alpha_{th}}, \quad \hat{m}^n = \frac{m^n}{1-\mu_m} \\ v^n &= \mu_v v^{n-1} + (1-\mu_v)\left(\frac{\partial \mathrm{RKL}(\alpha)}{\partial \alpha_{th}}\right)^2, \quad \hat{v}^n = \frac{v^n}{1-\mu_v}, \end{aligned} \tag{4.17}$$

where the hyperparameters $\mu_m$ and $\mu_v$ control the decay rate for these moving averages. Note that the gradient $\frac{\partial \mathrm{RKL}(\alpha)}{\partial \alpha_{th}}$ requires computing the partial derivatives of the RKL distance with respect to the threshold $\alpha_{th}$. In the following numerical experiments, this derivative is computed by numerical differentiation (forward differencing). Here, the factor $\sqrt{\hat{v}^k}$ is calculated based on the norm of the gradient and is responsible for scaling the momentum $\hat{m}^k$. In iterative updating using Eq. (4.16), the gradient is obtained by determining the value $\alpha_i$ for each particle $i$ using Eq. (4.15) for a given value $\alpha_{th}$ and calculating the RKL distance using Eq. (4.13). This procedure is summarized in Algorithm 4.1.

### 4.3.5 Iterative bias estimation with revised IEWPF idea

The need to select the $\alpha^n_{<1}$ or $\alpha^n_{\geq 1}$ branch and the existence of a gap between both branches is a limitation of the original IEWPF. However, the revised IEWPF [SEvLA19] solves this issue using the two-stage proposal method, as described in Section 2.1.7. Therefore, this section shows a method for estimating the bias component of the scalar factor $\alpha_i$ for the original IEWPF using the revised IEWPF concept.

**Algorithm 4.1** Iterative $\alpha_i$ Selection Algorithm

**Input:** Number of particles $N$, initial value $\alpha_{th}^{initial}$, maximum number of iterations $k_{\max}$, step-size factor $\lambda_\alpha$
**Output:** Scalar factor $\alpha_i$ for 1 to $N$ particles

```
Calculate α^n_{<1} and α^n_{≥1} solutions at time step n.
Initialize α_th: α^{k=1}_th = α^{initial}_th.
for k = 1, ..., k_max do
    for i = 1, ..., N do
        Determine α^k_i from α^k_th based on Eq. (4.15).
        Calculate z^n_i from Eq. (4.7).
    end for
    Calculate RKL^n(α^k_{1:N}) based on Eq. (4.12), Eq. (4.13), and Eq. (4.14).
    Update α^k_th by Eq. (4.16).
end for
Adopt α^{k_max}_{1:N} for α^n_{1:N}.
```

With reference to the revised IEWPF, instead of sampling from the posterior distribution in Eq. (4.7), we assume the following equation:

$$\begin{aligned} z_i^n &= \zeta_i^n + \alpha_0^{1/2} P^{1/2} \xi_i^n + \alpha_i^{1/2} P^{1/2} \xi_i^n, \\ &= \zeta_i^n + \left(\alpha_0^{1/2} + \alpha_i^{1/2}\right) P^{1/2} \xi_i^n, \end{aligned} \tag{4.18}$$

where $\alpha_0$ is a particle-independent scalar factor, called the bias. Note that the particle-dependent factor $\alpha_i$, like the revised IEWPF, adopts only the solution of the $\alpha_{\leq 1}^n$ branch. Therefore, it is not necessary here to decide which branch to select from, but only to estimate the value of $\alpha_0$.

First, as in Section 4.3.2, we define a distribution $q(z^n|\alpha_0)$ with $\alpha_0$ as the hyper-parameter. Then, in the same way as we assumed $\alpha_i = 1$ in Eq. (4.7), we define the distribution that assumes $\alpha_0^{1/2} + \alpha_i^{1/2} = 1$ as the target distribution $\hat{p}(z^n)$ for the RKL distance. Finally, similar to Section 4.3.4, $\alpha_0$ is estimated to minimize $\mathrm{RKL}(\alpha_0) = \mathrm{KL}[q(z^n|\alpha_0)||\hat{p}(z^n)]$. Thus, the update corresponding to Eq. (4.16) is as follows:

$$\alpha_0^{k+1} \leftarrow \alpha_0^k - \lambda_0 \frac{\hat{m}^k}{\sqrt{\hat{v}^k} + \delta_\alpha}, \tag{4.19}$$

**Algorithm 4.2** Iterative $\alpha_0$ Estimation Algorithm

**Input:** Number of particles $N$, initial value $\alpha_0^{initial}$, maximum number of iterations $k_{\max}$, step-size factor $\lambda_0$
**Output:** Particle-independent scalar factor $\alpha_0$

```
Calculate α_{≤1}^n solution at time step n.
Initialize α_th: α_0^{k=1} = α_0^{initial}.
for k = 1, ..., k_max do
    for i = 1, ..., N do
        Set α_0 as the estimate and α_i^n from α_{≤1}^n branch.
        Calculate z_i^n from Eq. (4.18).
    end for
    Calculate RKL^n(α_0^k) based on Eq. (4.12), Eq. (4.13), and Eq. (4.14).
    Update α_0^k by Eq. (4.19).
end for
Adopt α_0^{k_max} for α_0^n.
```

where $k$ is the iteration counter and $\lambda_0$ is the step-size factor. Then, the moving averages of the gradient $\hat{m}^k$ and the squared gradient $\hat{v}^k$ are calculated as follows:

$$\begin{aligned} m^n &= \mu_m m^{n-1} + (1-\mu_m)\frac{\partial \mathrm{RKL}(\alpha_0)}{\partial \alpha_0}, \quad \hat{m}^n = \frac{m^n}{1-\mu_m} \\ v^n &= \mu_v v^{n-1} + (1-\mu_v)\left(\frac{\partial \mathrm{RKL}(\alpha_0)}{\partial \alpha_0}\right)^2, \quad \hat{v}^n = \frac{v^n}{1-\mu_v}. \end{aligned} \tag{4.20}$$

Note that the gradient $\frac{\partial \mathrm{RKL}(\alpha_0)}{\partial \alpha_0}$ requires computing the partial derivatives of the RKL distance with respect to the factor $\alpha_0$. This procedure is summarized in Algorithm 4.2.

## 4.4 Numerical experiments

The validity of the present method is demonstrated through two synthetic test cases as follows. In the first case, we evaluate the method using an additive linear model of parameters, for which we can obtain an analytical solution. We then use the 1000-dimensional Lorenz 96 model [Lor96] to evaluate its applicability to high-dimensional,

nonlinear models.

### 4.4.1 Linear model case

To compare the estimates obtained by our new method with the true values calculated analytically, we use the same linear model presented below, as in the previous chapter.

$$\begin{aligned} z^n = \begin{pmatrix} x^n \\ \theta^{n-1} \end{pmatrix} &= \tilde{F} \begin{pmatrix} x^{n-1} \\ \theta^{n-2} \end{pmatrix} + \tilde{G} \begin{pmatrix} \beta^n \\ \eta^{n-1} \end{pmatrix} \\ &= \begin{pmatrix} F_x & F_{x\theta} \\ O & I \end{pmatrix} \begin{pmatrix} x^{n-1} \\ \theta^{n-2} \end{pmatrix} + \begin{pmatrix} I & F_{x\theta} \\ O & I \end{pmatrix} \begin{pmatrix} \beta^n \\ \eta^{n-1} \end{pmatrix}, \\ y^n &= \tilde{H} z^n + \epsilon^n, \end{aligned} \tag{4.21}$$

where $x \in \mathbb{R}^{N_x}$ is the model state vector with dimension $N_x = 1000$ and $\theta \in \mathbb{R}^{N_\theta}$ is the parameter vector with dimension $N_\theta = 1$. The matrices $F_x \in \mathbb{R}^{N_x \times N_x}$ and $F_{x\theta} \in \mathbb{R}^{N_x \times N_\theta}$ define a linear model. In the following, $F_x$=$I$ and all elements of $F_{x\theta}$ are set to 0.1: $F_{x\theta} = (0.1, 0.1, \dots)^T$. $\beta$ and $\eta$ are random perturbations following the model-error pdf $\mathcal{N}(0, Q_\beta)$ and parameter-error pdf $\mathcal{N}(0, Q_\eta)$, respectively. We assume that all variables are observed and $\epsilon$ is the observation error obeying the observation error pdf $\mathcal{N}(0, R)$. When the initial prior pdf is Gaussian, the true posterior pdf should also be Gaussian. When the posterior pdf at time step $n-1$ is Gaussian with covariance matrix $P_{n-1|n-1}$, the forecast covariance matrix $P_{n|n-1}$ for the prior pdf expressed in Eq. (4.21) can be calculated as follows:

$$P_{n|n-1} = \tilde{F} P_{n-1|n-1} \tilde{F}^T + \tilde{G} \tilde{Q} \tilde{G}^T, \tag{4.22}$$

where $\tilde{Q}$ expresses the covariance matrix for the augmented perturbation. Because we assume a time-independent state transition matrix $\tilde{F}$, the covariance matrix satisfying the linear system defined by Eq. (4.21) converges to the steady-state matrix $P$ such that $P_{n|n-1} = P_{n-1|n-2} \equiv P$, and satisfies the discrete-time Riccati equation [Won68] as follows:

$$P = \tilde{F} P \tilde{F}^T - \tilde{F} P \tilde{H}^T \left( \tilde{H} P \tilde{H}^T + R \right)^{-1} \tilde{H} P \tilde{F}^T + \tilde{G} \tilde{Q} \tilde{G}^T. \tag{4.23}$$

Then, the posterior distribution of the forecast covariance matrix can be obtained according to the Kalman filter equations given by

$$\begin{aligned} P_{n|n} &= P - K\tilde{H}P, \\ K &= P\tilde{H}^T \left(\tilde{H}P\tilde{H}^T + R\right)^{-1}. \end{aligned} \tag{4.24}$$

Therefore, the shape of the true posterior pdf in Eq. (4.21) is obtained numerically by solving Eq. (4.23) and compared with the distribution obtained from the new method.

The comparison procedure is as follows. First, synthetic data are generated. The initial ensemble members $z_i^0$ are sampled from the background error pdf $\mathcal{N}(0, B)$. One member from the ensemble generated from the model error and the background error pdf is used as the "truth." Then, observations are generated from this "truth" and the observation error is defined by the covariance matrix $R$. In the following experiments, the true value of the parameter is zero. The background error covariance matrix for the states $B_x$ and the parameter $B_\theta$ are chosen as a diagonal matrix with the main diagonal values 1 and 0, respectively. The true model error covariance matrix for the states $Q_\beta$ and the parameter $Q_\eta$ are chosen as a diagonal matrix with the main diagonal values 0.04 and 0, respectively. The observation error matrix $R$ is diagonal, and the main diagonal value is set to 0.01.

Then, data assimilation is performed using the generated observations. For the matrices $Q_\beta$, $B_x$, and $R$, the same respective matrices are chosen as when the observation was generated. The main diagonal values of matrices $Q_\eta$ and $B_\theta$ for parameters are set as 0.04 and 0.001, respectively. The number of particles is set to $N = 20$ to demonstrate the validity of the estimation with a few particles. Regarding observations, we consider the condition that all model state variables $x$ are observed at every step.

The following is about the verification of the two $\alpha_i$ estimation methods described above. First, we show an example of evaluating an iterative estimation method in one time step by each method. Then, we compare the variance of the estimated pdf with the analytical solution obtained from Eq. (4.23) and Eq. (4.24), and the solution sampled from the $\alpha \geq 1$ and $\alpha < 1$ branches.

### Branch selection method

Below are the evaluation results for the iterative branch selection method described in Section 4.3.4.

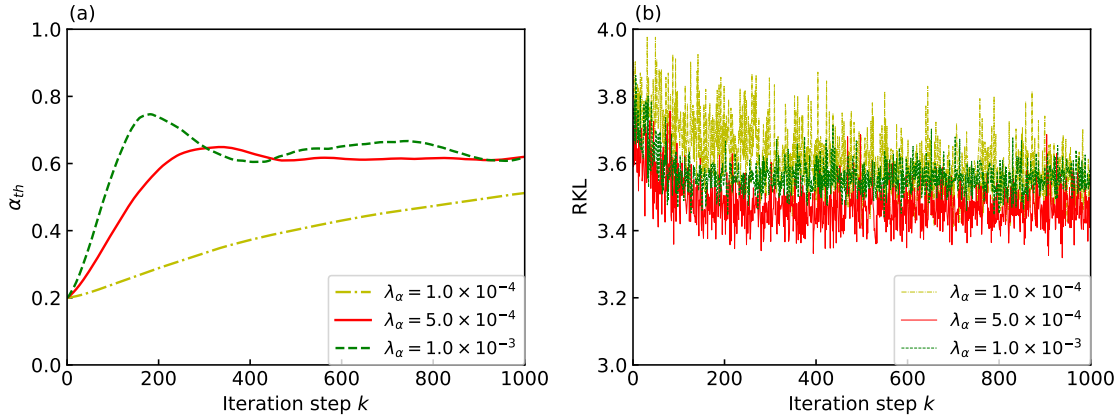


Figure 4.3: Comparison of three step-size factors $\lambda_\alpha = 1.0 \times 10^{-4}, 5.0 \times 10^{-4}$, and $1.0 \times 10^{-3}$ in iterative estimation method: (a) estimated result for $\alpha_{th}$ and (b) RKL value.

Figure 4.3 shows one process of the iterative estimation method described in Algorithm 4.1 at the initial time step and compares the dependence of the step-size factor $\lambda_\alpha$ defined in Eq. (4.16). The horizontal axis indicates the number of iterations, and for the vertical axis, (a) is the estimated $\alpha_{th}$ and (b) is the RKL value. The initial value $\alpha_{th}^{initial}$ is set to 0.2. The $\alpha_{th}$ value is underestimated when the step-size factor $\lambda_\alpha$ is $1.0 \times 10^{-4}$, while overshoot and oscillations are seen when $\lambda_\alpha$ is $1.0 \times 10^{-3}$. When $\lambda_\alpha$ is $5.0 \times 10^{-4}$, the RKL distance converges to a smaller value than in other cases. Therefore, in the following, the factor $\lambda_\alpha$ is set to $5.0 \times 10^{-4}$ and the number of iterations is set to $k_{max}$ =1000. Also, $\alpha_{th}^{k=1}$ is initialized only at the first time step (i.e., $n$ =1) and set to $\alpha_{th}^{k=1} \leftarrow \alpha_{th}^{k=k_{max}}$ at subsequent steps.

Figure 4.4 shows a histogram of the sampled or selected $\alpha_i$ accumulated from steps 20 to 1000. Graphs (a) $\alpha < 1$ and (c) $\alpha \geq 1$ are the results of sampling from branches $\alpha_{<1}^n$ and $\alpha_{\geq 1}^n$, respectively, based on the definition. Graph (b) in the middle is the result of the iterative branch selection method described in Algorithm 4.1. In this case, the time-averaged ratio of both branches $\overline{\alpha_{<1}} : \overline{\alpha_{\geq 1}}$ is 0.60 : 0.40, which is close to the 50%

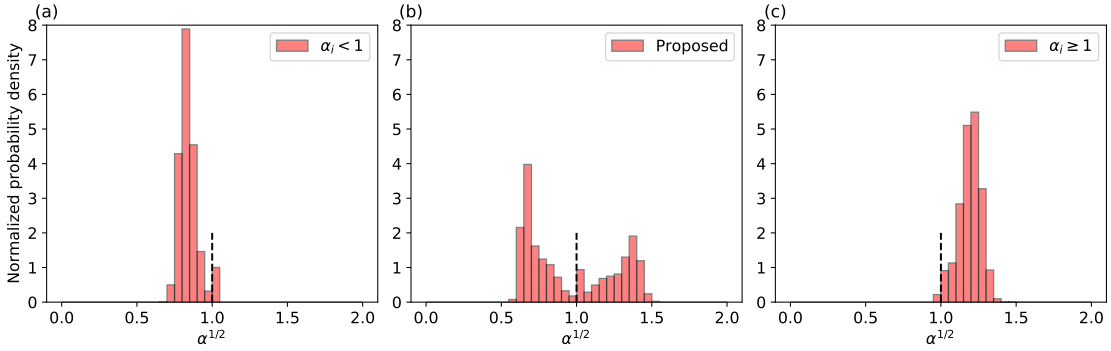


Figure 4.4: Histograms of $\alpha^{1/2}$ accumulated from steps 20 to 1000 to compare (a) the $\alpha < 1$ sampling case, (b) proposed branch selection method, and (c) the $\alpha \geq 1$ sampling case.

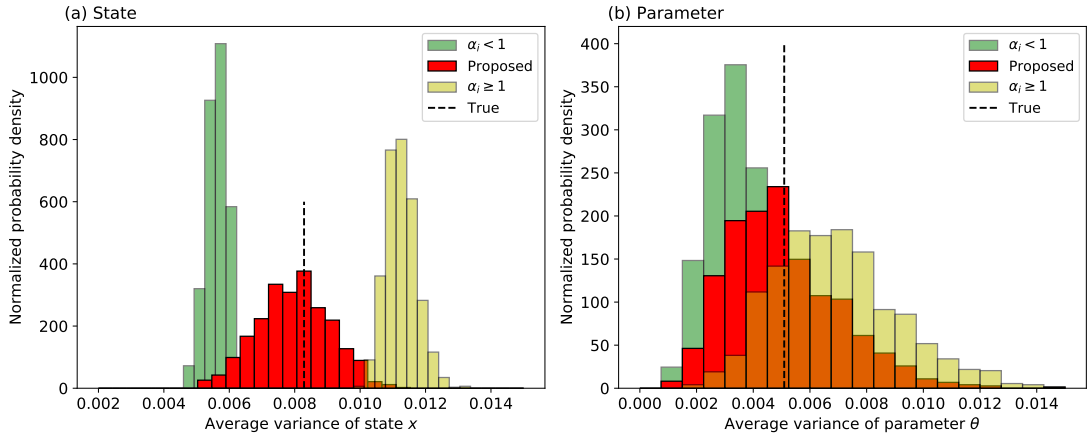


Figure 4.5: Histograms of variance accumulated from steps 20 to 1000 to compare present branch selection method and two sampling cases: $\alpha \geq 1$ and $\alpha < 1$. The graphs in (a) and (b) represent states and parameters, respectively, and a dashed line shows the true variance.

sampling assumption shown in [ZvLA16] and Chapter 3.

Figure 4.5 shows histograms of the variance accumulated from steps 20 to 1000 to compare the new branch selection method and the two sampling cases: $\alpha \geq 1$ and $\alpha < 1$. The variances of both (a) state $Var(x)$ and (b) parameter $Var(\theta)$ are averaged

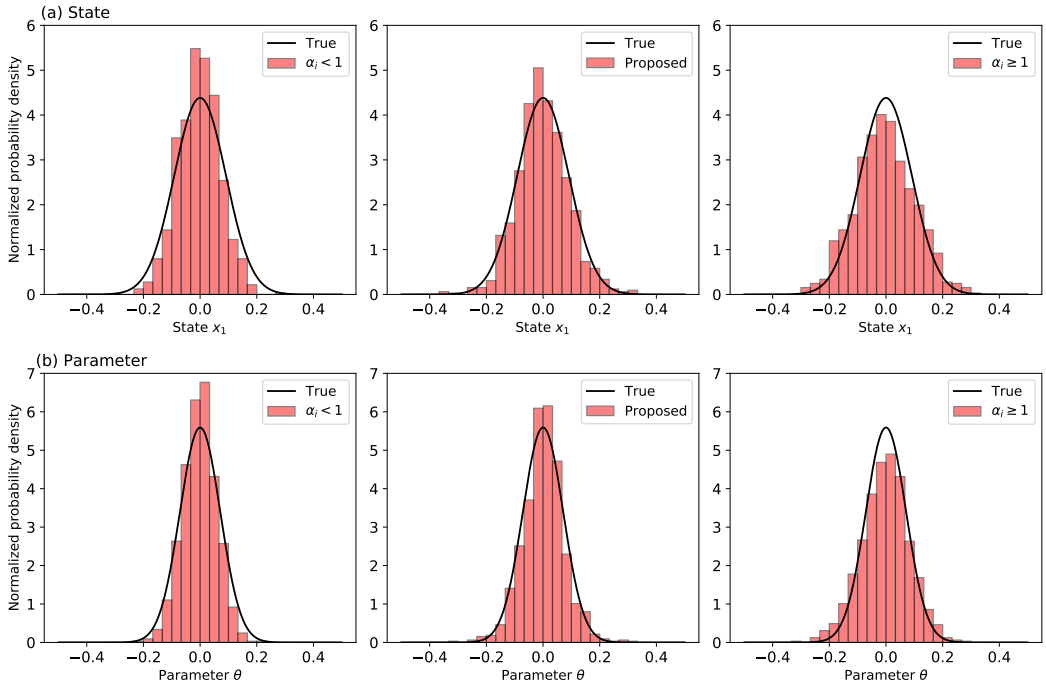


Figure 4.6: Posterior pdfs obtained from new branch selection method and two sampling cases: $\alpha \geq 1$ and $\alpha < 1$. The graphs in (a) and (b) represent the state $x_1$ and parameter $\theta$, respectively, and the solid curves represent the true pdf.

over the dimension and the number of particles as follows:

$$\begin{aligned} \overline{Var(x^n)} &= \frac{1}{N_x}\sum_{j=1}^{N_x}\frac{1}{N}\sum_{i=1}^{N}\left(x_i^n - \overline{x^n}\right)_j^2, \\ \overline{Var(\theta^n)} &= \frac{1}{N}\sum_{i=1}^{N}\left(\theta_i^n - \overline{\theta^n}\right)^2, \end{aligned} \tag{4.25}$$

where the index $j$ denotes the elements of the states $x$ and $N_x$, the dimension of the variable is $N_x = 1000$, the dimension of the parameter is $N_\theta = 1$, and the number of particles is $N = 20$. Also, $\overline{x^n}$ and $\overline{\theta^n}$ are the ensemble means. The true variances obtained from Eq. (4.24) are shown as "True." The results show that the variance of the states and parameters estimated by the new method is closer to the true value than when sampling from two branches.

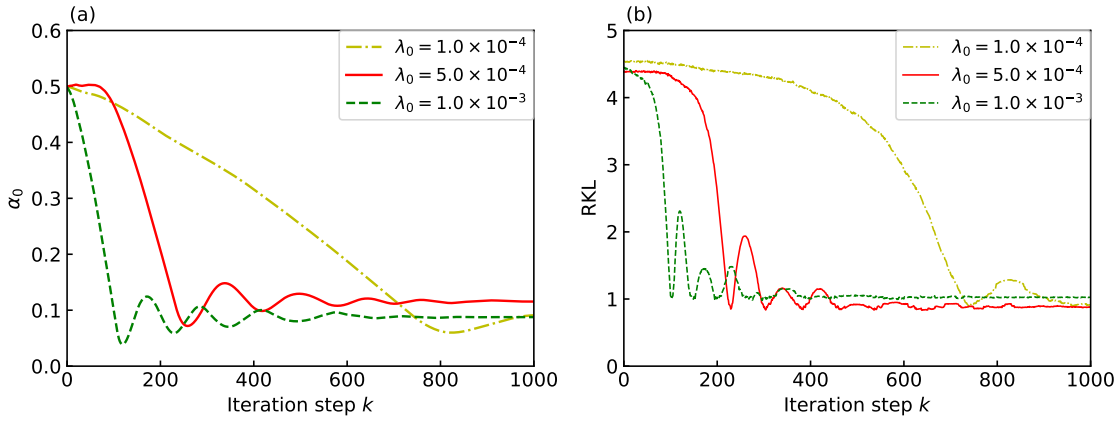


Figure 4.7: Comparison of three step-size factors $\lambda_0 = 1.0 \times 10^{-4}, 5.0 \times 10^{-4}$, and $1.0 \times 10^{-3}$ in iterative estimation method: (a) estimated result of $\alpha_0$ and (b) RKL value.

Figure 4.6 compares the posterior pdf obtained from the new branch selection method and two sampling cases with the true pdf. The graphs in (a) and (b) represent the variables and parameters, respectively, and the solid curves represent the true pdf. Because the ensemble size is too small compared to the number of model dimensions, each estimated pdf is displayed as a cumulative histogram with the average value subtracted at each step in steps 20 through 1000. The shape of the pdfs for variables and parameters estimated by the present branch selection method is close to the true shape.

### Bias estimation method

Below are the evaluation results of the iterative bias estimation method described in Section 4.3.5.

Figure 4.7 shows one process of the iterative estimation method described in Algorithm 4.2 at the initial time step and compares the dependence of the step-size factor $\lambda_0$ defined in Eq. (4.19). The horizontal axis indicates the number of iterations; for the vertical axis, (a) is the estimated $\alpha_0$ and (b) is the RKL value. The initial value $\alpha_{th}^{initial}$ is set to 0.5. From Figure 4.3, the case of $\lambda_0 = 5.0 \times 10^{-4}$ that converges to the smallest RKL value is used as the step-size factor $\alpha_0$ in the subsequent experiments. In the following, the number of iterations is set to $k_{max}$ =1000. Also, $\alpha_0^{k=1}$ is initialized only at

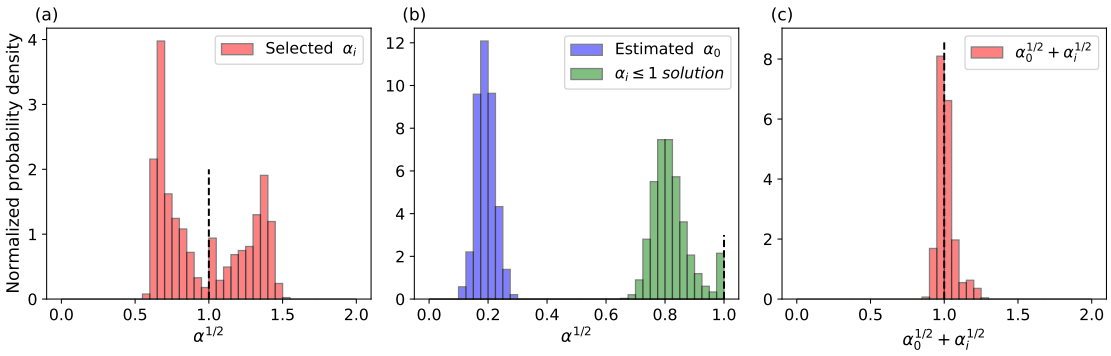


Figure 4.8: Comparison of histograms of $\alpha^{1/2}$ accumulated from steps 20 to 1000: (a) $\alpha_i$ selected by branch selection method, (b) $\alpha_0$ (blue) estimated by bias estimation method and $\alpha_i \leq 1$ solution (green), and (c) sum of estimated $\alpha_0$ and $\alpha_i \leq 1$ solution.

the first time step (i.e., $n$ =1) and set to $\alpha_0^{k=1} \leftarrow \alpha_0^{k=k_{max}}$ at subsequent steps.

Figure 4.8 shows histograms of the present bias estimation method accumulated from steps 20 to 1000, and compares them to the aforementioned (a) branch selection method. In Figure 4.8 (b), the histogram of the estimated $\alpha_0$ is the result of the iterative estimation method described in Algorithm 4.2. In addition, the histogram obtained as a solution to $\alpha_{\leq 1}^n$ is also shown. Then, Figure 4.8 (c) shows the histogram of the sum of the estimated $\alpha_0$ and $\alpha_{\leq 1}$ solution (i.e., $\alpha_0^{1/2} + \alpha_i^{1/2}$). It can be seen that the new method, which incorporates the concept of the revised IEWPF, is able to estimate the $\alpha_0$ value as a particle-independent bias value instead of using the solution of $\alpha_{\geq 1}^n$. As a result, compared to (a), the branch selection method, the histogram of (c) the bias estimation method has a narrower distribution with $\alpha_0^{1/2} + \alpha_i^{1/2} = 1$ at the top.

Figure 4.9 shows histograms of the variance accumulated from steps 20 to 1000 to compare the proposed bias estimation method and the two sampling cases: $\alpha \geq 1$ and $\alpha < 1$, as in Figure 4.5. Comparison with Figure 4.5 shows that the bias estimation method has a variance close to the true value and a smaller variance dispersion.

### 4.4.2 Lorenz 96 model case

As in the previous chapter, the Lorenz 96 model [Lor96] is used as the time evolution model to evaluate the validity of the new method in a nonlinear high-dimensional

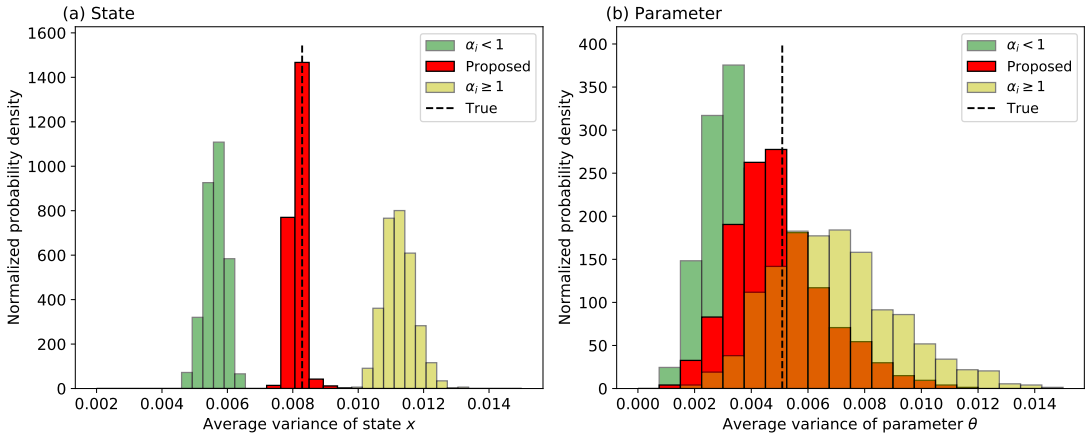


Figure 4.9: Histograms of variance accumulated from steps 20 to 1000 to compare new bias estimation method and two sampling cases: $\alpha \geq 1$ and $\alpha < 1$. The graphs in (a) and (b) represent variables and parameters, respectively, and dashed lines show the true variance.

system. Thus, the time evolution model is as follows:

$$\frac{d}{dt}x_j = (x_{j+1} - x_{j-2})x_{j-1} - x_j + F_j, \tag{4.26}$$

where index $j = 1, \ldots, N_x$ with cyclic indices, $N_x$ is the total number of dimensions, and $F_j$ is as follows:

$$F_j(\theta_0, \theta_1, \theta_2) = c_0\theta_0 + c_1\theta_1 \sin\left(\frac{2\pi}{c_2\theta_2}j\right), \tag{4.27}$$

for which $c_0$, $c_1$, and $c_2$ are true values and $\theta_0$, $\theta_1$, and $\theta_2$ are their respective scale parameters, which have to be estimated. The values $c_0$ and $c_1$ are set to 8 and 4, respectively, and $c_2$ is set to the same value as the dimension of the model state $N_x$.

The procedure for the following experiment is the same as for the previous linear model. The true model error covariance matrix $Q_\beta$ for states is chosen as a tridiagonal matrix, the main diagonal values being 0.10 and both sub- and super-diagonal values being 0.025. The background error covariance matrix for the state $B_x$ and the parameter $B_\theta$ are chosen as a diagonal matrix with the main diagonal values 1 and 0, respectively. The true observation error matrix $R$ is diagonal, with main diagonal val-

ues of 0.02. For the assimilation, we choose the same matrices $Q_\beta$ and $B_x$ for the state equivalent to when the observation was generated. The matrices $Q_\eta$ and $B_\theta$ for parameters are diagonal matrices with main values of $5.0 \times 10^{-6}$ and 0.001, respectively. The dimension $N_x$ is set to 1000, and the number of particles is set to $N = 20$, which is the same setting as in the linear model experiment. The observation condition considers that all model states are observed every fourth step (i.e., the assimilation interval is 4). Figure 4.10 compares the assimilation results up to step 1500 for state $x_1$ and the

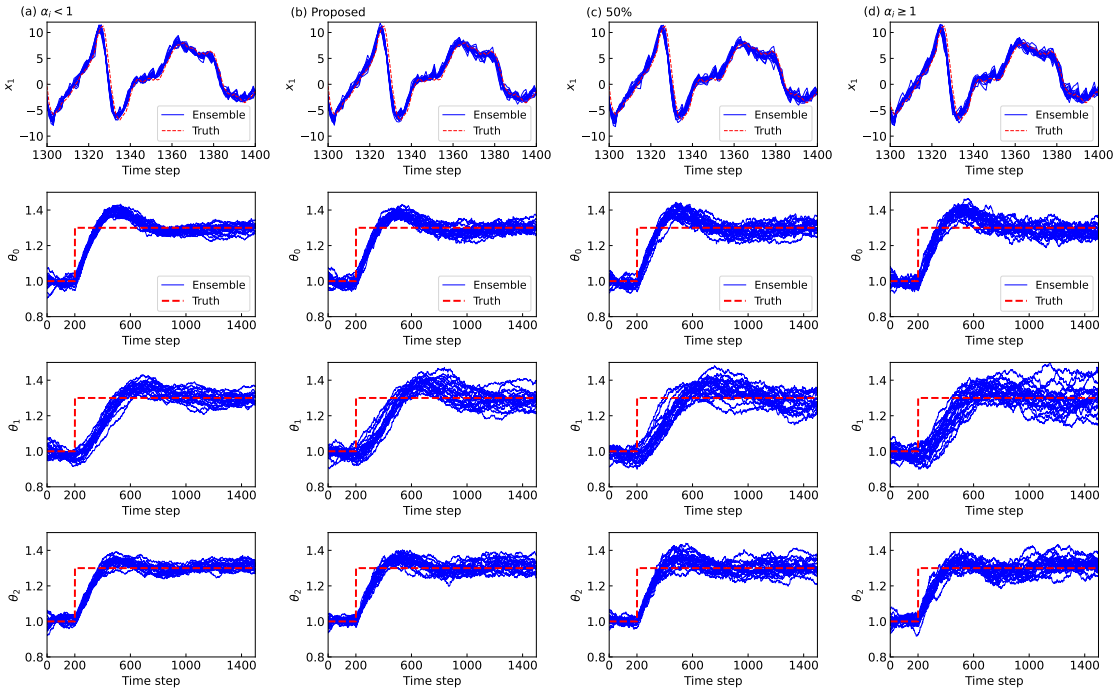


Figure 4.10: Comparison of estimated state and parameter trajectories between (a) $\alpha < 1$, (b) new branch selection method, (c) 50% each from both branches, and (d) $\alpha \geq 1$. The solid curves (blue) show each of the 20 ensemble members, and the dashed lines (red) show the true parameter value. Only steps 1300–1400 are shown for the state, and each true parameter is increased by 30% at step 200.

three scale parameters $\theta_0, \theta_1$, and $\theta_2$ with the three sampling cases ($\alpha < 1$, $\alpha \geq 1$, and 50% each from both branches) and the present branch selection method. The true parameters are increased by 30% each at step 200 to evaluate time-varying parameter estimation. All cases show that the particles follow the true value after the parameter

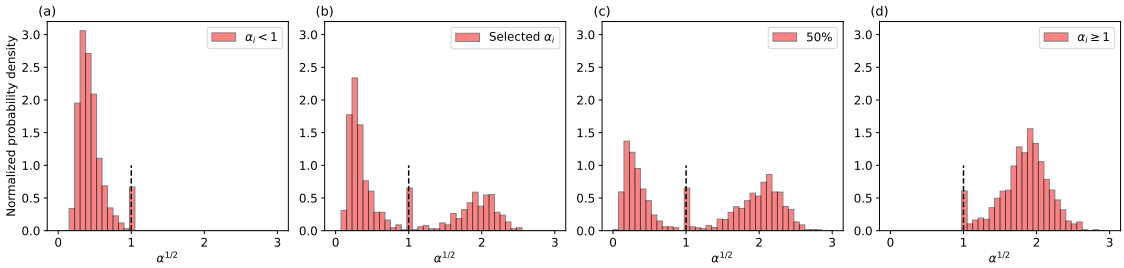


Figure 4.11: Histograms of $\alpha^{1/2}$ accumulated from steps 20 to 200 to compare (a) sampling case $\alpha < 1$, (b) proposed branch selection method, (c) the sampling case with 50% each from both branches, and (d) the sampling case with $\alpha \geq 1$.

change, but there is a difference in the particle spread. In the following, we first show the period before the parameter change (i.e., 200 steps or less), then the period after the parameter change (i.e., after 200 steps).

### Before parameter change

First, the results of the branch selection method described in Section 4.3.4 are presented. Figure 4.11 shows histograms of the sampled and selected $\alpha_i$ accumulated from steps 20 to 200. Graphs (a) $\alpha < 1$, (c) 50%, and (d) $\alpha \geq 1$ are the results of sampling from branches $\alpha^n_{<1}$ and / or $\alpha^n_{\geq 1}$. Graph (b) with "Selected $\alpha_i$" is the result of the iterative branch selection method described in Algorithm 4.1. Compared to the case of the linear model shown in Figure 4.4, it is noticeable that the solutions of the branch $\alpha^n_{\geq 1}$ in each method are spread over larger values. This means that the log-weight offsets $c_i$ are large, which may be attributed to the nonlinearity of the function and estimation error. Comparison of the results of the 50% sampling with those of the new branch selection method shows that the time-averaged ratio of $\overline{\alpha_{<1}}$ to $\overline{\alpha_{\geq 1}}$ in the new method is 0.67 to 0.33, which, unlike the linear model, deviates from 50%. This means that more branches of $\alpha^n_{<1}$ (i.e., the small $\alpha$ value) are selected, and the solution of this branch $\alpha^n_{<1}$ narrows the distribution. Thus, it can be seen that choosing $\alpha$ to be close to the distribution with $\alpha = 1$ results in the distribution being narrower than that of the 50% sampling.

Next, the results of the bias estimation method described in Section 4.3.5 are pre-

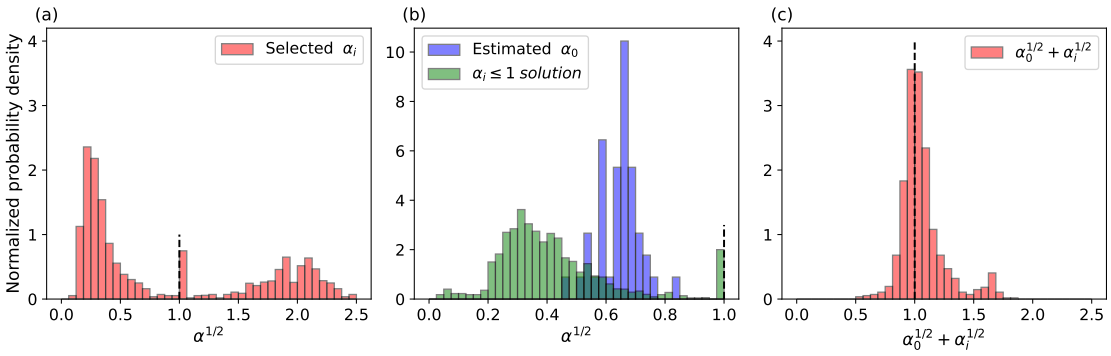


Figure 4.12: Comparison of histograms of $\alpha^{1/2}$ accumulated from steps 20 to 200: (a) $\alpha_i$ selected by branch selection method, (b) $\alpha_0$ (blue) estimated by bias estimation method and $\alpha \leq 1$ solution (green), and (c) sum of estimated $\alpha_0$ and $\alpha \leq 1$ solution.

sented. Figure 4.12 shows histograms of $\alpha_0$ estimated by the bias estimation method accumulated from steps 20 to 200 before the abrupt parameter change. Comparing with the histogram of $\alpha_i \leq 1$ solution, we see that the distribution of $\alpha_0$ is shifted to the right so that the variance is not underestimated. In other words, $\alpha_0$ suppresses the under-dispersion caused by using only $\alpha_i \leq 1$ solution. Compared with (a) $\alpha_i$ selected by the branch selection method in the original IEWPF, the histogram of (c) $\alpha_0^{1/2} + \alpha_i^{1/2}$ has a narrower distribution with 1.0 at the top.

Figure 4.13 compares the RMSE and particle spread for different $\alpha$ solutions. Note that $\alpha_0$ estimation and $\alpha_i$ selection indicate the results for the bias estimation and branch selection methods, respectively. Each box plot in the graphs shows the time-averaged RMSE and spread at forecast and filtering steps 20–200, before the abrupt parameter change. The IQR of the box plot indicates the dispersion across the dimensions of the model states (1000) and parameters (3). From the result for the states shown in Figure 4.13 (a), we can see that the difference in RMSE for different $\alpha$ solutions is almost negligible, while there is a more noticeable difference in the spread. In the results for $\alpha_0$ estimation and $\alpha_i$ selection, the RMSE and spread values are close, meaning their ratio is close to one. In general, the ratio of the RMSE and spread is evaluated as a performance indicator for ensemble forecasting, and this ratio should become one for Gaussian variables. This is due to the desirability that the RMSE, which represents the forecast error, and the spread, which represents the uncertainty of the forecast, should

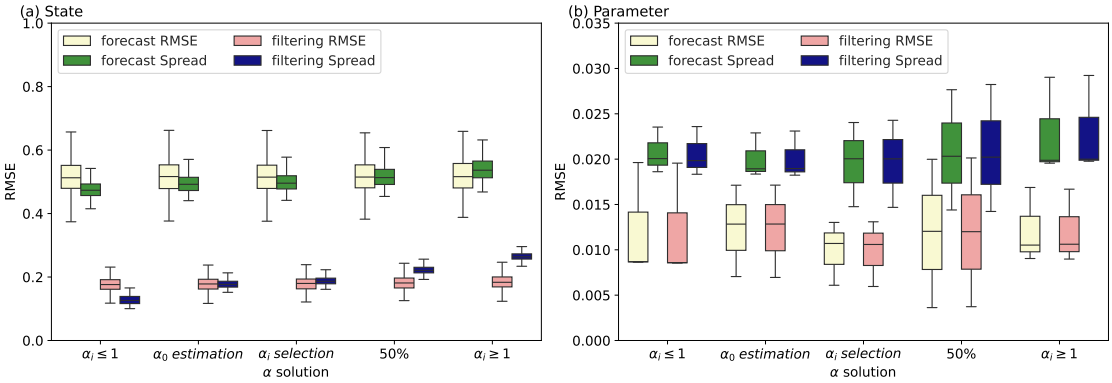


Figure 4.13: Box plots showing comparisons of RMSE and spread for forecast and filtered ensembles between different $\alpha$ solutions: $\alpha < 1$, $\alpha_0$ estimation, $\alpha_i$ selection, 50% sampling, and $\alpha \geq 1$. Each IQR indicates the dispersion of the (a) state and (b) parameter elements averaged over the forecast and filtering steps 20–200. Outliers are not plotted.

be of equal magnitude [FAAT14]. From the result for the parameters shown in Figure 4.13 (b), the spread in $\alpha_0$ estimation and $\alpha_i$ selection is smaller than that of the 50% sampling. Thus, the smaller spread in $\alpha_i$ selection is due to the fact that more solutions in the $\alpha^n_{<1}$ branch are selected, as shown in Figure 4.11. In contrast, the smaller spread in the $\alpha_0$ estimation is due to the distribution of $\alpha$ with $\alpha = 1$ as the vertex, as shown in Figure 4.12.

### After parameter change

Below are the results after the parameter change (i.e., 200 steps or more). The results of the branch selection method are shown first. Figure 4.14 shows histograms of the sampled and selected $\alpha^{1/2}$ accumulated from steps 200 to 1500. Compared to the results before the parameter change shown in Figure 4.11, it can be seen that the distribution of $\alpha_{<1}$ branch is approaching zero. On the other hand, the value of $\alpha$ has increased further in cases (b), (c), and (d), which include $\alpha_{\geq 1}$ branch. This is thought to be due to an increase in the log-weight offsets as a result of an increase in the estimation error due to parameter changes. However, the time-averaged ratio of $\overline{\alpha_{<1}}$ to $\overline{\alpha_{\geq 1}}$ in the branch selection method is 0.68 to 0.32, and there is no significant change compared

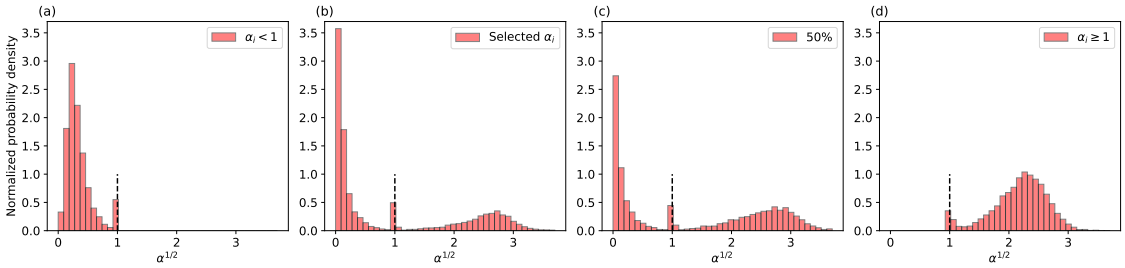


Figure 4.14: Histograms of $\alpha^{1/2}$ accumulated from steps 200 to 1500 to compare (a) sampling case $\alpha < 1$, (b) new branch selection method, (c) sampling case of 50% each from both branches, and (d) sampling case $\alpha \geq 1$.

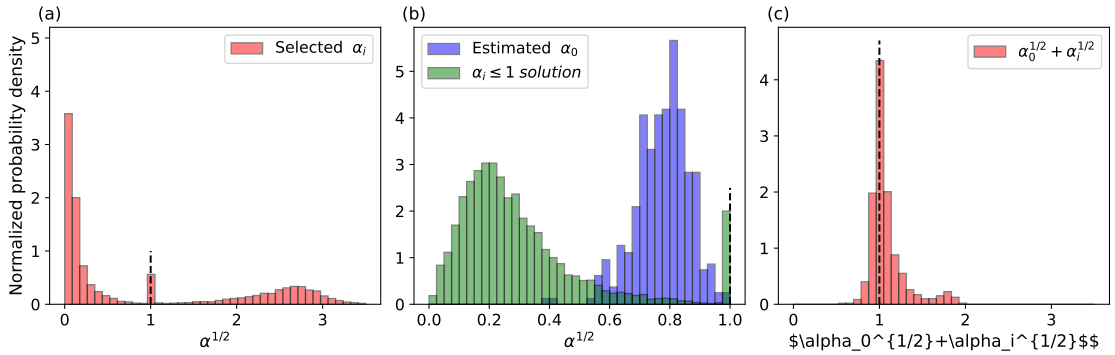


Figure 4.15: Comparison of histograms of $\alpha^{1/2}$ accumulated from steps 200 to 1500: (a) $\alpha_i$ selected by branch selection method, (b) $\alpha_0$ estimated by bias estimation method and $\alpha \leq 1$ solution, and (c) sum of estimated $\alpha_0$ and $\alpha \leq 1$ solution.

to the ratio before the parameter change. In other words, even when the estimation error is increased, the selection ratio of the two branches remains the same.

Next, the results of the bias estimation method are shown. Figure 4.15 shows histograms of the $\alpha_0$ estimation method accumulated from steps 200 to 1500 after the abrupt parameter change. Compared to Figure 4.12 (b) before the parameter change, the distribution of $\alpha_0$ is shifted further to the right because the distribution of $\alpha \leq 1$ solution is even closer to zero. As a result, the histogram of (c) $\alpha_0^{1/2} + \alpha_i^{1/2}$ has a distribution with 1.0 at the top, as it did before the parameter change.

Figure 4.16 shows the time-averaged RMSE and spread at the forecast and filtering

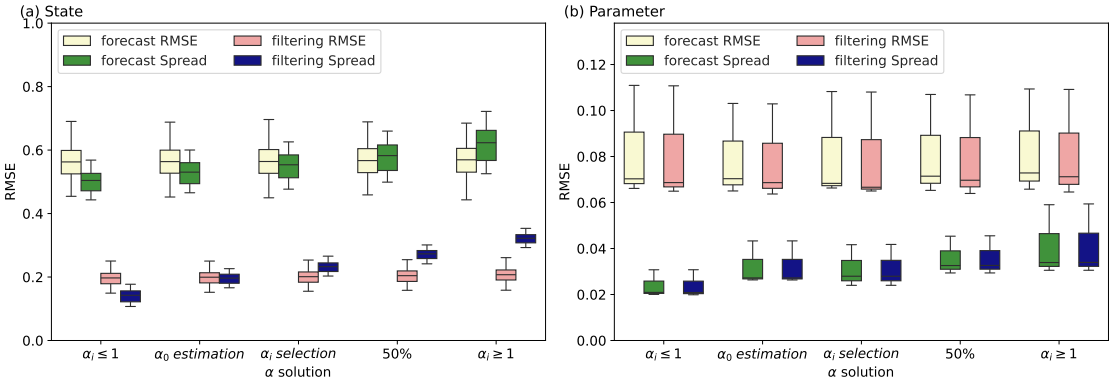


Figure 4.16: Box plots showing comparisons of RMSE and spread for forecast and filtered ensembles between different $\alpha$ solutions: $\alpha < 1$, $\alpha_0$ estimation, $\alpha_i$ selection, 50% sampling, and $\alpha \geq 1$. Each IQR indicates the dispersion of the (a) state and (b) parameter elements averaged over forecast and filtering steps 200–1500.

steps in the same way as Figure 4.13. Note that the time-averaged range is 200–1500 steps. This evaluates the accuracy between the parameter change and when the estimated value approaches the true value. From the result for the states shown in Figure 4.16 (a), as before the parameter change, there is almost no difference in the RMSE, but the difference in the spread is larger for both forecast and filtering. From the result for the parameters shown in Figure 4.13 (b), the spread of the $\alpha_0$ estimation and $\alpha_i$ selection is shifted to a smaller value compared to 50% sampling. Also, the RMSE in the $\alpha_0$ estimation tends to be slightly smaller. In other words, both the $\alpha_i$ selection and $\alpha_0$ estimation methods achieve equal or better performance without prior assumptions about $\alpha$ (e.g., 50% sampling).

## 4.5 Discussion

The evaluation results using the linear and nonlinear Lorenz 96 models described above suggest the following points. First, for the branch selection method, we see that even in the linear model, the time-averaged ratio of the branch $\overline{\alpha_{<1}}$ is greater than 50%, and even more in the nonlinear model. From this, it follows that the increase in the log-weight offsets due to the nonlinearity of the model influences the selection of the scalar

Table 4.1: Comparison of the time and particle mean values for $\overline{\alpha^{1/2}}$ in linear and Lorenz 96 models with 50% sampling, $\alpha_i$ selection, and $\alpha_0$ estimation methods.

| Model | 50% | $\alpha_i$ selection | $\alpha_0$ estimation |
|---|---|---|---|
| Linear | 1.02 | 0.960 | 1.01 |
| Lorenz 96 (before) | 1.22 | 0.891 | 1.06 |
| Lorenz 96 (after) | 1.32 | 0.924 | 1.09 |

factor $\alpha_i$. Specifically, an increase in the log-weight offsets increases the value of $\alpha$ in the $\alpha^n_{\geq 1}$ branch that spreads the distribution. Then, in order to reduce the KL distance from the distribution with $\alpha_i = 1$, the selection ratio of the $\alpha^n_{\leq 1}$ branch, which narrows the distribution, increases. As a result, the ratio of the RMSE to the spread can be closer to one compared to the case where the ratio is fixed at 50%. However, the relationship between the scalar factor $\alpha$ and the model nonlinearity requires further investigation, such as an evaluation using a linear model that does not include parameter estimation.

Next, we discuss the differences between the branch selection and bias estimation methods. As shown in Figure 4.8 (a) and Figure 4.5, the new $\alpha_i$ selection method with the original IEWPF obtained appropriate variance (i.e., particle spread) by mixing the $\alpha < 1$ and $\alpha \geq 1$ branch solutions. However, as shown in Figure 4.8 (b) and Figure 4.9, the new $\alpha_0$ estimation method, which incorporates the revised IEWPF idea, obtains the appropriate variance by adding the particle-independent bias $\alpha_0$ and the particle-dependent factor $\alpha_i$. In other words, the dispersion of the variance was reduced by not using the $\alpha \geq 1$ branch solution. That is, it is not affected by the gap between the two branches of the solution $\alpha_i$ due to the weight offsets to equalize the weights of each particle. Therefore, the factor $\alpha_i$ does not spread to large values, as shown in the Lorenz 96 model example above. In fact, the evaluation results for the Lorenz 96 model confirmed that it works as well as the linear model when parameters change abruptly. Note that it still contains $\alpha = 1$ solutions; that is, this particle has no degrees of freedom. It has been pointed out that this one degree of freedom less than the number of particles may be acceptable for very high-dimensional systems, but it leads to bias [vLKN+19]. Therefore, constructing the factor for each particle to not include the $\alpha_i = 1$ solution may be a future study.

Next is a quantitative comparison of the $\alpha^{1/2}$ values determined by each method.

Define the time and particle mean of $\alpha^{1/2}$ as follows:

$$\overline{\alpha^{1/2}} = \frac{1}{T}\sum_{n=1}^{T}\frac{1}{N}\sum_{i=1}^{N}\left(\alpha_i^n\right)^{1/2}. \tag{4.28}$$

Table 4.1 compares the $\overline{\alpha^{1/2}}$ in the linear and Lorenz 96 models with the 50% sampling, $\alpha_i$ selection, and $\alpha_0$ estimation methods. The results for the Lorenz 96 model are shown separately before and after the abrupt parameter change. The mean value of $\alpha^{1/2}$ in the 50% sampling method is close to one in the linear model, although it does not assume a target distribution $\hat{p}(z^n)$ with $\alpha = 1$. However, it becomes larger than one in the nonlinear Lorenz 96 model, and even greater with the parameter change. In contrast, the $\alpha_i$ selection and $\alpha_0$ estimation methods minimize the KL divergence with the $\alpha = 1$ target distribution, so the mean value of $\alpha^{1/2}$ is close to one for both the linear and nonlinear Lorenz 96 models. In other words, it is less sensitive to model nonlinearity and parameter errors. Therefore, the method presented in this chapter allows for resilient estimation of time-varying parameters compared to existing IEWPFs applied to parameter estimation.

Finally, we discuss the validity of estimating the distribution to be close to the distribution with $\alpha_i = 1$. Assuming that the posterior distribution of the IEWPF given by Eq. (4.4) is Gaussian, the distribution $q\left(z^n|\alpha\right)$ that minimizes the KL distance can be expressed as follows:

$$q\left(z^n|\alpha_i\right) = \mathcal{N}\left(\zeta_i^n, \alpha_i P\right). \tag{4.29}$$

Similarly, we assume that the target distribution for KL minimization with $\alpha_i = 1$ is also normal:

$$\hat{p}\left(z^n\right) = \mathcal{N}\left(\zeta_i^n, P\right). \tag{4.30}$$

Then, the KL distance between the normal distributions $q\left(z^n|\alpha\right)$ and $\hat{p}\left(z^n\right)$ is given by

$$\begin{aligned}\mathrm{KL}[q(z^n|\alpha_i)||\hat{p}(z^n)] &= \int_{-\infty}^{\infty} q(z^n|\alpha_i)\frac{q(z^n|\alpha_i)}{\hat{p}(z^n)}dz \\ &= \log\left(\alpha_i^{-1/2}\right) + \frac{1}{2}\alpha_i - \frac{1}{2}.\end{aligned} \tag{4.31}$$

However, Eq. (4.5) that $\alpha_i$ must satisfy to equalize the weights is given by, under higher-

dimensional approximation (see [ZvLA16] for details):

$$\xi_i^{nT}\xi_i^n\alpha_i + 2N_x \log \alpha_i^{-1/2} - \xi_i^{nT}\xi_i^n - c_i = 0. \tag{4.32}$$

Here, because $\xi_i^{nT}\xi_i^n \sim N_x$, Eq. (4.32) can be rewritten as

$$\log\left(\alpha_i^{-1/2}\right) + \frac{1}{2}\alpha_i - \frac{1}{2} - \frac{c_i}{2N_x} = 0. \tag{4.33}$$

Thus, the following relationship can be obtained from Eq. (4.31) and Eq. (4.33):

$$\frac{c_i}{2N_x} = \mathrm{KL}[q(z^n|\alpha_i)||\hat{p}(z^n)]. \tag{4.34}$$

From Eq. (4.34), it can be seen that minimization of the log-weight offsets $c_i$ is equivalent to minimization of the KL distance. Therefore, the method of minimizing the KL distance to the distribution with $\alpha_i = 1$ is a reasonable method for determining the scalar factor $\alpha$.

## 4.6 Conclusion

This chapter described a method for incorporating IEWPF into a sequential estimation method for the scalar factor $\alpha$ that determines the variance of the posterior distribution. We assumed that the value of $\alpha_i$ for each particle $i$ is determined so that the KL distance from the optimal proposal distribution is minimized, that is, when all particles take $\alpha_i = 1$. Then, by converting the particle distribution into a histogram, it became possible to minimize the KL distance with a low computational load. As a result, the method can be applied to high-dimensional models and parameter estimation, and it achieved performance equivalent to or better than the results in Chapter 3 without making any prior assumptions about factor $\alpha$.

From evaluating the 1000-dimensional linear model with an unknown parameter, we confirmed that the variance value for the estimated posterior distribution was almost the same as the analytical value. Also, from evaluating the 1000-dimensional nonlinear Lorenz 96 model with three time-varying parameters, we confirmed that the same or better accuracy and ensemble quality (i.e., the ratio of the RMSE and spread)

can be obtained. The above results demonstrate the validity of our assumptions based on numerical evaluation, but theoretical proof is a subject for future research.

Finally, the method for estimating particle-independent bias values without using the $\alpha \geq 1$ solution, which introduces the idea of the revised IEWPF, was presented. The advantage of this method is that it can be realized using the same algorithm as the above method of selecting a solution for $\alpha_i$ based on KL minimization. In other words, the method based on KL minimization described in this chapter is versatile. Evaluation with the 1000-dimensional linear model confirms that this method has a variance close to the true value and a smaller variance dispersion. From the evaluation with the 1000-dimensional nonlinear Lorenz 96 model, both $\alpha_i$ selection and $\alpha_0$ estimation methods achieve equal or better performance without prior assumptions about $\alpha$ (e.g., 50% sampling). Therefore, the method presented in this chapter allows for a more resilient estimation of time-varying parameters than the existing IEWPF applied to parameter estimation.

# 5

# Conclusion and future work

## 5.1 Conclusion

This study investigated resilient estimation methods for states and time-varying parameters applicable to geophysical, climatological, and other high-dimensional applications. Estimating time-varying parameters plays an important role not only in improving prediction accuracy but also in determining when model characteristics change abruptly. To achieve this aim, we used the implicit equal-weights particle filter (IEWPF), which can prevent degeneracy by equalizing the weights of all particles. First, we combined the parameter vector with the state vector of the IEWPF using an augmented state space model with a correlated covariance matrix. This allowed for the estimation of sequential time-varying parameters in a high-dimensional nonlinear model. This satisfies requirements 1 and 2 described in Section 1.2.

We then introduced an IEWPF-based method that incorporates a nudging technique inspired by optimization algorithms in machine learning into the parameter time evolution model by using the flexibility of the proposal density in particle filtering. This improved the tracking performance for abrupt parameter changes and reduced the estimation accuracy difference for each parameter. This satisfies requirement 3 described in Section 1.2.

Finally, for the coefficient specific to the IEWPF that determines the particle distribution, we proposed an adaptive determination method using analytical solutions of the Lambert W function and iterative computation with the Kullback-Leibler (KL) divergence. This method makes it unnecessary to give the pre-set values. Furthermore, when compared to the original method, it was found that this method is more resilient to model nonlinearity and parameter errors. This satisfies requirement 4 described in Section 1.2.

## 5.2 Future work

The application of particle filters to high-dimensional nonlinear models is one of the major research topics in terms of suppression of degeneracy and computational complexity. The modified IEWPF focused on in this thesis is suitable for solving high-dimensional models with an extremely small number of particles, that is, with limited computational resources, but challenges remain.

Chapter 3 described an online estimation method that performs estimation each time observation data are available. Therefore, in a real application, the estimation, including forecasting and filtering, should be completed by the time the next observation data are acquired. As an example, one IEWPF step for the 10000-dimensional Lorenz 96 model takes approximately 9.5 minutes for a 20-particle parallel computation [BW15] with an Intel Core i9-7940X CPU at 3.1 GHz. A list of improvements to be considered for the current method is

- Ensemble approximation of matrix calculations
- Efficient method of computing parameter gradients
- Further parallelization and other implementation innovations

Note that the above computation time estimates were based on the simple Lorenz 96 model as a time evolution model but depend on the model when used in real applications. Chapter 3 also assumed a linear observation model and Gaussian error as the case in which analytical solutions are available. If we do not make the above assumptions, we have to obtain $\zeta_i^n$ in Eq. (3.20) by minimization of $-\log q\left(z^n|z_{1:N}^{n-1}, y^n\right)$, similar to, for example, the three-dimensional variational scheme. Also, because the Lambert W function cannot be used in general, the solution $\alpha_i$ to Eq. (3.25) must be obtained numerically. Thus, for general problems, the above points cause an increase in computational complexity. It is also necessary to evaluate the case where a method other than Adam, which was adopted as the optimization algorithm, is replaced with the latest optimizer. For example, if [PYHZ24] can be applied, this method eliminates the tuning of the step-size factor, which was a hyperparameter in this thesis.

In Chapter 4, the method described for determining factor $\alpha$ was validated by assuming that the KL distance between the posterior distribution and the optimal proposal distribution is minimized. However, this assumption has not been proven except under the assumption of a normal distribution. In addition, from the perspective of eliminating the need for prior assumptions and tuning, it is necessary to provide guidelines for setting the hyperparameters that determine the probability distribution in the IEWPF other than the factor $\alpha$. Specifically, the factor for parameter nudging and the diagonal values of the parameter-error covariance matrix are involved. For example, offline maximum likelihood estimation methods are expected to be applied.

The issue of time-varying parameter estimation is one of the larger research questions, and there are many aspects that can still be investigated. Because the method presented here has only been validated with the Lorenz 96 model, the applicability of the proposed method to various nonlinear problems in data assimilation should be investigated. Also, the time-varying pattern of the parameters was assumed to be the case where abrupt (staircase-like) changes occur simultaneously for all parameters in this thesis. In particular, it should be verified whether the first-order approximation of the parameters introduced in Eq. (3.3) account for the nonlinearity associated with the parameters. Therefore, more realistic time-varying patterns should be assumed and evaluated depending on the application and model.

Furthermore, the observation density, e.g., the dimension and frequency of observations, should be considered for real applications. The partially observed experiment described in Section 3.3.2 showed that the difference in estimation accuracy between parameters increases as the observation density decreases. Therefore, one possible countermeasure is to extract and estimate parameters that significantly impact estimation accuracy; for example, sensitivity analysis can be applied. Alternatively, some approaches investigate the impact of observations or find observations that contribute to improving estimation accuracy. For example, an observing system simulation experiment (e.g., ensemble-based method [TAFI07]), which builds a virtual observation system, could be useful.

# Bibliography

[ADST04] Christophe Andrieu, Arnaud Doucet, Sumeetpal S. Singh, and Vladislav B. Tadic, Particle methods for change detection, system identification, and control, Proceedings of the IEEE **92** (2004), no. 3, 423–438, https://doi.org/10.1109/JPROC.2003.823142.

[ADT05] Christophe Andrieu, Arnaud Doucet, and Vladislav B. Tadic, On-line parameter estimation in general state-space models, Proceedings of the 44th IEEE Conference on Decision and Control, IEEE, 2005, https://doi.org/10.1109/CDC.2005.1582177, pp. 332–337.

[AT11] Jaison Thomas Ambadan and Youmin Tang, Sigma-point particle filter for parameter estimation in a multiplicative noise environment, Journal of Advances in Modeling Earth Systems **3** (2011), no. 4, https://doi.org/10.1029/2011MS000065.

[ATY+18] Md Zahangir Alom, Tarek M. Taha, Christopher Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Brian C. Van Esesn, Abdul A. S. Awwal, and Vijayan K. Asari, The history began from alexnet: A comprehensive survey on deep learning approaches, arXiv preprint arXiv:1803.01164 (2018).

[AvL13] Melanie Ades and Peter Jan van Leeuwen, An exploration of the equivalent weights particle filter, Quarterly Journal of the Royal Meteorological Society **139** (2013), no. 672, 820–840, https://doi.org/10.1002/qj.1995.

[AvL15] ———, The equivalent-weights particle filter in a high-dimensional system, Quarterly Journal of the Royal Meteorological Society **141** (2015), no. 687, 484–503, https://doi.org/10.1002/qj.2370.

[Bay63] Thomas Bayes, LII. an essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, FRS communicated by Mr. Price, in a letter to John Canton, AMFR S, Philosophical transactions of the Royal Society of London (1763), no. 53, 370–418.

[BEM01] Craig H. Bishop, Brian J. Etherton, and Sharanya J. Majumdar, Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects, Monthly weather review **129** (2001), no. 3, 420–436, https://doi.org/10.1175/1520-0493(2001)129

[BKM17] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe, Variational inference: A review for statisticians, Journal of the American statistical Association **112** (2017), no. 518, 859–877.

[Bot10] Léon Bottou, Large-scale machine learning with stochastic gradient descent, Proceedings of COMPSTAT'2010: 19th International Conference on Computational StatisticsParis France, August 22-27, 2010 Keynote, Invited and Contributed Papers, Springer, 2010, https://doi.org/10.1007/978-3-7908-2604-3_16, pp. 177–186.

[BSN03] Thomas Bengtsson, Chris Snyder, and Doug Nychka, Toward a nonlinear ensemble filter for high-dimensional systems, Journal of Geophysical Research: Atmospheres **108** (2003), no. D24, https://doi.org/10.1029/2002JD002900.

[BW15] Philip A. Browne and Simon Wilson, A simple method for integrating a complex model into an ensemble data assimilation system using MPI, Environmental Modelling & Software **68** (2015), 122–128, https://doi.org/10.1016/j.envsoft.2015.02.003.

[CBBE18] Alberto Carrassi, Marc Bocquet, Laurent Bertino, and Geir Evensen, Data assimilation in the geosciences: An overview of methods, issues, and

perspectives, Wiley Interdisciplinary Reviews: Climate Change **9** (2018), no. 5, e535, https://doi.org/10.1002/wcc.535.

[CGH+96] Robert M. Corless, Gaston H. Gonnet, David E.G. Hare, David J. Jeffrey, and Donald E. Knuth, On the Lambert W function, Advances in Computational mathematics **5** (1996), 329–359.

[CGM07] Olivier Cappé, Simon J. Godsill, and Eric Moulines, An overview of existing methods and recent advances in sequential Monte Carlo, Proceedings of the IEEE **95** (2007), no. 5, 899–924, https://doi.org/10.1109/JPROC.2007.893250.

[CLB13] Adam M. Clayton, Andrew C. Lorenc, and Dale M. Barker, Operational implementation of a hybrid ensemble/4D-Var global data assimilation system at the Met Office, Quarterly Journal of the Royal Meteorological Society **139** (2013), no. 675, 1445–1461, https://doi.org/10.1002/qj.2054.

[CLSH18] Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong, On the convergence of a class of Adam-type algorithms for non-convex optimization, arXiv preprint arXiv:1808.02941 (2018).

[CMT10] Alexandre Chorin, Matthias Morzfeld, and Xuemin Tu, Implicit particle filters for data assimilation, Communications in Applied Mathematics and Computational Science **5** (2010), no. 2, 221–240, http://dx.doi.org/10.2140/camcos.2010.5.221.

[CP18] Matthew Cooper and Tristan Perez, Dual-particle-filtering for recursive estimation of agricultural-machinery dynamics, IFAC-PapersOnLine **51** (2018), no. 15, 658–663, https://doi.org/10.1016/j.ifacol.2018.09.210.

[DDFG+01] Arnaud Doucet, Nando De Freitas, Neil James Gordon, et al., Sequential Monte Carlo methods in practice, vol. 1, Springer, New York, USA, 2001, https://doi.org/10.1007/978-1-4757-3437-9.

[DLG+16] Chao Deng, Pan Liu, Shenglian Guo, Zejun Li, and Dingbao Wang, Identification of hydrological model parameter variation using ensemble

Kalman filter, Hydrology and Earth System Sciences **20** (2016), no. 12, 4949–4961, https://doi.org/10.5194/hess-20-4949-2016.

[DT03] Arnaud Doucet and Vladislav B. Tadić, Parameter estimation in general state-space models using particle methods, Annals of the institute of Statistical Mathematics **55** (2003), 409–422, https://doi.org/10.1007/BF02530508.

[EDS98] Geir Evensen, Dick P. Dee, and Jens Schröter, Parameter estimation in dynamical models, Ocean modeling and parameterization (1998), 373–398.

[Eve94] Geir Evensen, Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics, Journal of Geophysical Research: Oceans **99** (1994), no. C5, 10143–10162.

[EVvL22] Geir Evensen, Femke C. Vossepoel, and Peter Jan van Leeuwen, Data assimilation fundamentals: A unified formulation of the state and parameter estimation problem, 2022, https://doi.org/10.1007/978-3-030-96709-3.

[FAAT14] V. Fortin, M. Abaza, F. Anctil, and R. Turcotte, Why should ensemble spread match the RMSE of the ensemble mean?, Journal of Hydrometeorology **15** (2014), no. 4, 1708–1713, https://doi.org/10.1175/JHM-D-14-0008.1.

[Fu15] Michael C. Fu, Stochastic gradient estimation, Springer, New York, USA, 2015, https://doi.org/10.1007/978-1-4939-1384-8_5.

[Han15] Lauren A. Hannah, Stochastic optimization, International Encyclopedia of the Social & Behavioral Sciences **2** (2015), 473–481.

[Hig97] Tomoyuki Higuchi, Monte Carlo filter using the genetic algorithm operators, Journal of Statistical Computation and Simulation **59** (1997), no. 1, 1–23.

[HM98] Peter L. Houtekamer and Herschel L. Mitchell, Data assimilation using an ensemble Kalman filter technique, Monthly weather review **126** (1998), no. 3, 796–811.

[HS00] Thomas M. Hamill and Chris Snyder, A hybrid ensemble Kalman filter–3D variational analysis scheme, Monthly Weather Review **128** (2000), no. 8, 2905–2919, https://doi.org/10.1175/1520-0493(2000)128

[KB14] Diederik P. Kingma and Jimmy Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).

[KDS+15] Nikolas Kantas, Arnaud Doucet, Sumeetpal S. Singh, Jan Maciejowski, and Nicolas Chopin, On particle methods for parameter estimation in state-space models, Statistical Science **30** (2015), no. 3, 328 – 351, https://doi.org/10.1214/14-STS511.

[Kit96] Genshiro Kitagawa, Monte Carlo filter and smoother for non-Gaussian nonlinear state space models, Journal of computational and graphical statistics **5** (1996), no. 1, 1–25.

[Kit98] ———, A self-organizing state-space model, Journal of the American Statistical Association (1998), 1203–1215.

[Kiv03] G.A. Kivman, Sequential parameter estimation for stochastic systems, Nonlinear Processes in Geophysics **10** (2003), no. 3, 253–259, https://doi.org/10.5194/npg-10-253-2003.

[Lor96] Edward N. Lorenz, Predictability: A problem partly solved, Proc. Seminar on predictability, vol. 1, Reading, Cambridge University Press, 1996.

[LS83] Lennart Ljung and Torsten Söderström, Theory and practice of recursive identification, MIT press, Cambridge, UK, 1983.

[Mee01] Alistair I. Mees, Nonlinear dynamics and statistics, Springer Science & Business Media, New York, USA, 2001, https://doi.org/10.1007/978-1-4612-0177-9.

[MTAC12] Matthias Morzfeld, Xuemin Tu, Ethan Atkins, and Alexandre J. Chorin, A random map implementation of implicit filters, Journal of Computational Physics **231** (2012), no. 4, 2049–2066, https://doi.org/10.1016/j.jcp.2011.11.022.

[NUH07] Shin'ya Nakano, Genta Ueno, and Tomoyuki Higuchi, Merging particle filter for sequential data assimilation, Nonlinear Processes in Geophysics **14** (2007), no. 4, 395–408, https://doi.org/10.5194/npg-14-395-2007.

[PMM+93] T.N. Palmer, F. Molteni, R. Mureau, R. Buizza, P. Chapelet, and J. Tribbia, Ensemble prediction, Proc. ECMWF Seminar on Validation of models over Europe, vol. 1, 1993, pp. 21–66.

[PVMM18] Sara Pérez-Vieites, Inés P. Mariño, and Joaquín Míguez, Probabilistic scheme for joint parameter estimation and state prediction in complex dynamical systems, Physical Review E **98** (2018), no. 6, 063305, https://doi.org/10.1103/PhysRevE.98.063305.

[PYHZ24] Yijiang Pang, Shuyang Yu, Bao Hoang, and Jiayu Zhou, Towards stability of parameter-free optimization, arXiv preprint arXiv:2405.04376 (2024).

[RC15] Sebastian Reich and Colin Cotter, Probabilistic forecasting and Bayesian data assimilation, Cambridge University Press, Cambridge, UK, 2015.

[RKK19] Sashank J Reddi, Satyen Kale, and Sanjiv Kumar, On the convergence of Adam and beyond, arXiv preprint arXiv:1904.09237 (2019).

[RPM13] Juan Jose Ruiz, Manuel Pulido, and Takemasa Miyoshi, Estimating model parameters with ensemble-based data assimilation: A review, Journal of the Meteorological Society of Japan. Ser. II **91** (2013), no. 2, 79–99, https://doi.org/10.2151/jmsj.2013-201.

[Rud16] Sebastian Ruder, An overview of gradient descent optimization algorithms, arXiv preprint arXiv:1609.04747 (2016).

[SBBA08] Chris Snyder, Thomas Bengtsson, Peter Bickel, and Jeff Anderson, Obstacles to high-dimensional particle filtering, Monthly Weather Review **136** (2008), no. 12, 4629–4640, https://doi.org/10.1175/2008MWR2529.1.

[SBM15] Chris Snyder, Thomas Bengtsson, and Mathias Morzfeld, Performance bounds for particle filters using the optimal proposal, Monthly Weather Review **143** (2015), no. 11, 4750–4761, https://doi.org/10.1175/MWR-D-15-0144.1.

[SCZZ19] Shiliang Sun, Zehui Cao, Han Zhu, and Jing Zhao, A survey of optimization methods from a machine learning perspective, IEEE transactions on cybernetics **50** (2019), no. 8, 3668–3681, https://doi.org/10.1109/TCYB.2019.2950779.

[SEvLA19] Jacob Skauvold, Jo Eidsvik, Peter Jan van Leeuwen, and Javier Amezcua, A revised implicit equal-weights particle filter, Quarterly Journal of the Royal Meteorological Society **145** (2019), no. 721, 1490–1502, https://doi.org/10.1002/qj.3506.

[SJ15] Naratip Santitissadeekorn and Christopher Jones, Two-stage filtering for joint state-parameter estimation, Monthly Weather Review **143** (2015), no. 6, 2028–2042, https://doi.org/10.1175/MWR-D-14-00176.1.

[SvLN24] Mineto Satoh, Peter Jan van Leeuwen, and Shin'ya Nakano, Online state and time-varying parameter estimation using the implicit equal-weights particle filter, Quarterly Journal of the Royal Meteorological Society (2024), https://doi.org/10.1002/qj.4698.

[TAFI07] David GH Tan, Erik Andersson, Michael Fisher, and Lars Isaksen, Observing-system impact assessment using a data assimilation ensemble technique: application to the ADM–Aeolus wind profiling mission, Quarterly Journal of the Royal Meteorological Society: A journal of the atmospheric sciences, applied meteorolo gy and physical oceanography **133** (2007), no. 623, 381–390, https://doi.org/10.1002/qj.43.

[TC87] Olivier Talagrand and Philippe Courtier, Variational assimilation of meteorological observations with the adjoint vorticity equation. I: Theory, Quarterly Journal of the Royal Meteorological Society **113** (1987), no. 478, 1311–1328.

[vL03] Peter Jan van Leeuwen, Nonlinear ensemble data assimilation for the ocean, Seminar on recent developments in data assimilation for atmosphere and ocean, ECMWF, 2003.

[vL09] ———, Particle filtering in geophysical systems, Monthly Weather Review **137** (2009), no. 12, 4089–4114, https://doi.org/10.1175/2009MWR2835.1.

[vL10] ———, Nonlinear data assimilation in geosciences: an extremely efficient particle filter, Quarterly Journal of the Royal Meteorological Society **136** (2010), no. 653, 1991–1999, https://doi.org/10.1002/qj.699.

[vLKN+19] Peter Jan van Leeuwen, Hans R. Künsch, Lars Nerger, Roland Potthast, and Sebastian Reich, Particle filters for high-dimensional geoscience applications: A review, Quarterly Journal of the Royal Meteorological Society **145** (2019), no. 723, 2335–2365, https://doi.org/10.1002/qj.3551.

[VvL07] Femke C. Vossepoel and Peter Jan van Leeuwen, Parameter estimation using a particle method: Inferring mixing coefficients from sea level observations, Monthly weather review **135** (2007), no. 3, 1006–1020, https://doi.org/10.1175/MWR3328.1.

[Won68] William M. Wonham, On a matrix Riccati equation of stochastic control, SIAM Journal on Control **6** (1968), no. 4, 681–697.

[ZMC+17] Zhiliang Zhu, Zhiqiang Meng, Tingting Cao, Zhengjiang Zhang, and Yuxing Dai, Particle filter-based robust state and parameter estimation for nonlinear process systems with variable parameters, Measurement Science and Technology **28** (2017), no. 6, 065003, https://doi.org/10.1088/1361-6501/aa5dc9.

[ZvLA16] Mengbin Zhu, Peter Jan van Leeuwen, and Javier Amezcua, Implicit equal-weights particle filter, Quarterly Journal of the Royal Meteorological Society **142** (2016), no. 698, 1904–1919, https://doi.org/10.1002/qj.2784.