Sequential Estimation of States and Parameters of Nonlinear State Space Models Using Particle Filter and Natural Evolution Strategy

1st Yoshiki Kobayashi
School of Computing
Tokyo Institute of Technology
Yokohama, Japan
Email: kobayashi.y.bx@m.titech.ac.jp

2nd Isao Ono
School of Computing
Tokyo Institute of Technology
Yokohama, Japan
Email: isao@c.titech.ac.jp

***Abstract*—This paper proposes a new sequential estimation method for simultaneously estimating states and parameters of a state space model. Particle filter (PF) is known as a method that can estimate states in difficult sequential state estimation problems with nonlinearity and non-Gaussianity. PF updates an ensemble consisting of multiple particles representing states of a state space model in order to estimate the true state, based on observation, at each time step. However, when PF estimates not only states but also parameters of the state space model at the same time, it is observed that the estimation accuracy deteriorates. When estimating both states and parameters, PF utilizes particles representing states and particles. In order to overcome the problem of PF, we propose a new method that sequentially estimates states by PF and parameters by the separable natural evolution strategy (SNES). SNES is one of the most powerful black-box function optimization methods. In order to confirm the effectiveness of the proposed method, we compare the performance of the proposed method and that of PF using two nonlinear state space models, the Van der Pol model and the Lorenz model. In the Van der Pol model, the median MSE values of the state and the parameter of the proposed method were 0.003610 and 0.01468 and those of PF were 4.228 and 6.520, respectively. In the Lorenz model, the median MSE values of the state and the parameter of the proposed method were 0.002639 and 0.003479 and those of PF were 309.5 and 1.470, respectively. The smaller MSE is, the better the performance is.**

***Index Terms*—Nonlinear state space model, State estimation, Parameter estimation, Particle filter, Natural evolution strategy**

I. Introduction

The sequential state estimation problem is an important problem that appears in a wide range of fields such as meteorology, oceanography and robotics. The sequential state estimation problem is based on a state space model consisting of two equations, a state equation representing the state of the system and an observation equation representing the observation process of the state. The state equation and the observation equation contain noise. If the state and observation equations of a model are both linear, the model is called the linear model; otherwise, it is called the nonlinear model. If the noise of the state equation and/or the observation one follows the normal distribution, the model is called the Gaussian model; otherwise it is called the non-Gaussian model. Nonlinear models and non-Gaussian ones are more difficult to estimate than linear ones.

In sequential state estimation problems, it is sometimes necessary to estimate not only states but also parameters that are the coefficients of the state equations simultaneously, which makes the estimation more difficult. Simultaneous estimation of states and parameters is an important problem because it appears in a wide range of fields such as parameter estimation of internal resistance in lithium ion batteries [1], that of tracked vehicle slip models [2] and that of hydraulic conductivity in water flow models [3].

There have been proposed several promising sequential state estimation methods for linear/non-linear and Gaussian/non-Gaussian models. The Kalman filter (KF) [4] is for linear and Gaussian models. The extended Kalman filter [5] (EKF), the ensemble Kalman filter (EnKF) [6] and the unscented Kalman filter (UKF) [7] can be applied to nonlinear and Gaussian models. The particle filter (PF) [8] is a method that is applicable to nonlinear models and non-Gaussian models.

PF is a powerful sequential state estimation method known to show good estimation accuracy in nonlinear and non-Gaussian models. PF expresses the probability distribution of the true state using an ensemble composed of multiple particles representing states. PF sequentially estimates the true states at each time step by updating the ensemble. However, PF has a serious problem in that the estimation accuracy deteriorates when estimating parameter, i.e. coeficients, of the state equation in addition to the states.

In this paper, in order to overcome the problem of PF, we propose a new method that sequentially estimates parameters and states by separately updating the ensemble of the states and the probability distribution of parameter every time step. The proposed method updates the state ensemble and the probability distribution of parameters by PF and the natural evolution strategy (NES) [9], [10], respectively. NES is one of the most powerful black-box function optimization methods. NES searches for the optimal hyper-parameters of a probability distribution which minimizes the expected evaluation value of an objective function by using the natural gradient method

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

[11], instead of directly minimizing the evaluation value of the objective function. In order to show the effectiveness of the proposed method, we compare the performance of the proposed method and that of PF through some experiments.

Section II explains the sequential state estimation problem. In Section III, we introduce PF as a conventional method and point out its problem. In order to remedy the problem of PF, we propose a new method that updates the probability distribution of parameter by NES in Section IV. Section V shows the experimental results where the proposed method outperformed PF on two famous benchmark problems. In Section VI, we discuss the behavior of the proposed method and PF by investigating the time transition diagram of the estimated states and parameters of the proposed method and PF. Section VII is a summary and future work.

II. Sequential State Estimation Problem

The sequential state estimation problem is a problem of estimating the state of a dynamic system from observation online. The states are estimated from observations obtained sequentially because the true states cannot be directly observed. The dynamic system is represented by a state-space model. The state-space model is composed of a state equation and an observation equation. The state equation and the observation equation are given by (1) and (2), respectively.

$$\boldsymbol{x}_t = \boldsymbol{f}\left(\boldsymbol{x}_{t-1}, \boldsymbol{a}, \boldsymbol{v}_t\right), \tag{1}$$
$$\boldsymbol{y}_t = \boldsymbol{g}\left(\boldsymbol{x}_t\right) + \boldsymbol{w}_t, \tag{2}$$

where $\boldsymbol{x}_t \in \mathbb{R}^{d_{\mathrm{s}}}$ is the state of the dynamic system at time $t \in \mathbb{Z}$, $\boldsymbol{y}_t \in \mathbb{R}^{d_{\mathrm{m}}}$ is the observation at time t, $\boldsymbol{a} \in \mathbb{R}^{d_{\mathrm{p}}}$ is the parameter of the state equation, d_{s} is the dimension of the state, d_{p} is the dimension of the parameter, and $\boldsymbol{v}_t$ and $\boldsymbol{w}_t$ are system noise and observation noise, respectively. The system noise represents the stochastic fluctuation of the dynamic system and the observation noise represents the noise generated in the observation process. If $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$ of a state-space model are both linear functions, the model is called the linear model; otherwise, it is called the nonlinear model. If $p_{\mathrm{s}}(\boldsymbol{v})$ and $p_{\mathrm{m}}(\boldsymbol{w})$ of a state-space model follow the normal distribution, the model is called the Gaussian model; otherwise it is called the non-Gaussian model. Nonlinear or non-Gaussian models are more difficult to estimate than linear and Gaussian models. Furthermore, when the parameter in addition to the state is required to be estimated at the same time, the estimation becomes more difficult.

III. Conventional Method And its Problem

A. Particle Filter (PF)

The particle filter (PF) [8], [12] is one of the most powerful sequential state estimation methods that show good estimation accuracy for nonlinear models and non-Gaussian models. PF approximates the probability distribution of a state with an ensemble. The ensemble $\{\boldsymbol{x}_t^{(i)}\}_{i=1}^N$ is a set of particles $\boldsymbol{x}_t^{(i)}$ that represents a state. The PF sequentially updates the ensemble by repeating the prediction step, the observation step, the filtering step and the step of calculating the estimation of the true state from time 1 to time T. The ensemble after the prediction step is called the prediction ensemble, and the ensemble after the filtering step is called the filtering ensemble. In the prediction step at time t, PF calculates a prediction ensemble $\{\boldsymbol{x}_{t|t-1}^{(i)}\}_{i=1}^N$ by substituting the filtering ensemble $\{\boldsymbol{x}_{t-1|t-1}^{(i)}\}_{i=1}^N$ of time $t-1$ to the state equation given by (3).

$$\boldsymbol{x}_{t|t-1}^{(i)} = \boldsymbol{f}\left(\boldsymbol{x}_{t-1|t-1}^{(i)}, \boldsymbol{a}, \boldsymbol{v}_t^{(i)}\right). \tag{3}$$

Next, in the observation step, PF obtains an observation $\boldsymbol{y}_t$ and calculates the likelihood $\lambda(\boldsymbol{x}_{t|t-1}^{(i)}) \in \mathbb{R}$ of each particle by (4).

$$\lambda\left(\boldsymbol{x}_{t|t-1}^{(i)}\right) = p\left(\boldsymbol{y}_t \mid \boldsymbol{x}_{t|t-1}^{(i)}\right), \tag{4}$$

where $p\left(\boldsymbol{y} \mid \boldsymbol{x}\right)$ is a likelihood function. Next, in the filtering step, PF performs resampling to obtain a filtering ensemble $\{\boldsymbol{x}_{t|t}^{(i)}\}_{i=1}^N$ at time t. In resampling, N particles are chosen from the prediction ensemble $\{\boldsymbol{x}_{t|t-1}^{(i)}\}_{i=1}^N$ with a probability proportional to the likelihood based on the observation, and a filtering ensemble $\{\boldsymbol{x}_{t|t}^{(i)}\}_{i=1}^N$ is obtained. Finally, PF calculates and outputs the estimation $\hat{\boldsymbol{x}}_{t|t}$ of the true state. In this paper, the estimation is the mean of the filtering ensemble [13].

Algorithm 1 PF

Require: $p_{\mathrm{s}}(\cdot), \boldsymbol{f}(\cdot), \boldsymbol{a}, p(\cdot), N, T$.
Input: $\{\boldsymbol{x}_{0|0}^{(i)}\}_{i=1}^N$.
Output: $\hat{\boldsymbol{x}}_{t|t} (t = 1, \cdots, T)$.
1: **for** $t = 1, \cdots, T$ **do**
2: **for** $i = 1, \cdots, N$ **do**
3: Generate $\boldsymbol{v}_t^{(i)} \sim p_{\mathrm{s}}(\boldsymbol{v})$.
4: $\boldsymbol{x}_{t|t-1}^{(i)} = \boldsymbol{f}(\boldsymbol{x}_{t-1|t-1}^{(i)}, \boldsymbol{a}, \boldsymbol{v}_t^{(i)})$.
5: **end for**
6: Observe $\boldsymbol{y}_t$.
7: **for** $i = 1, \cdots, N$ **do**
8: $\lambda(\boldsymbol{x}_{t|t-1}^{(i)}) = p(\boldsymbol{y}_t \mid \boldsymbol{x}_{t|t-1}^{(i)})$.
9: **end for**
10: **for** $i = 1, \cdots, N$ **do**
11: Choose a particle $\boldsymbol{x}_{t|t}^{(i)}$ from $\{\boldsymbol{x}_{t|t-1}^{(k)}\}_{k=1}^N$ with replacement with probabilities propotional to $\{\lambda(\boldsymbol{x}_{t|t-1}^{(k)})\}_{k=1}^N$.
12: **end for**
13: $\hat{\boldsymbol{x}}_{t|t} = \frac{1}{N}\sum_{i=1}^N \boldsymbol{x}_{t-1|t-1}^{(i)}$
14: **end for**

Algorithm 1 shows the algorithm of PF. PF requires the items in the "Require" line; a probability distribution $p_{\mathrm{s}}(\cdot)$ following a system noise, a function $\boldsymbol{f}(\cdot)$ of the state equation, a parameter $\boldsymbol{a}$ of the state equation, a likelihood function $p(\cdot)$, the number of particles N, and a final time T. PF takes the item in the "Input" line; a filtering ensemble $\{\boldsymbol{x}_{0|0}^{(i)}\}_{i=1}^N$ at time $t = 0$. PF outputs the estimations of the true states $\hat{\boldsymbol{x}}_{t|t} (t = 1, \cdots, T)$ in the "Output" line. State estimation from time 1 to T is performed in line 1 to 14. The prediction is performed in line 2 to 5. Line 3 generates a system noise $\boldsymbol{v}_t^{(i)} \sim p_s(\boldsymbol{v})$, and line 4 calculates a particle $\boldsymbol{x}_{t|t-1}^{(i)}$ of the prediction on ensemble. In line 6, the observation $\boldsymbol{y}_t$ is obtained. In line 7 to 9, the likelihood $\lambda(\boldsymbol{x}_{t|t-1}^{(i)})$ of each particle is calculated by the likelihood function $p(\boldsymbol{y}_t \mid \boldsymbol{x}_{t|t-1}^{(i)})$. In line

10 to 12, resampling is performed. In resampling, particles are extracted from the prediction ensemble $\{\boldsymbol{x}^{(i)}_{t|t-1}\}^N_{i=1}$ with a probability proportional to the likelihood $\{\lambda(\boldsymbol{x}^{(i)}_{t|t-1})\}^N_{i=1}$, and a filtering ensemble $\{\boldsymbol{x}^{(i)}_{t|t}\}^N_{i=1}$ is obtained. In line 13, the estimation of the true state is calculated.

If PF estimates a parameter $\boldsymbol{a}$ in addition to a state $\boldsymbol{x}$, a new state $\bar{\boldsymbol{x}}$ consisting of $\boldsymbol{x}$ and $\boldsymbol{a}$ is defined as in (5) [14].

$$\bar{\boldsymbol{x}}_t \leftarrow \begin{pmatrix} \boldsymbol{x}_t \\ \boldsymbol{a}_t \end{pmatrix}. \tag{5}$$

Then, prediction is performed using a model in which the state equation is changed from (3) to (6) using a function $\bar{\boldsymbol{f}}$ that returns the newly defined state $\bar{\boldsymbol{x}}^{(i)}_{t|t-1}$.

$$\begin{aligned}\bar{\boldsymbol{x}}^{(i)}_{t|t-1} &= \bar{\boldsymbol{f}}\left(\boldsymbol{x}^{(i)}_{t-1|t-1}, \boldsymbol{a}^{(i)}_{t-1}, \boldsymbol{v}^{(i)}_t\right), \\ &= \begin{pmatrix} \boldsymbol{f}\left(\boldsymbol{x}^{(i)}_{t-1|t-1}, \boldsymbol{a}^{(i)}_{t-1}, \boldsymbol{v}^{(i),\boldsymbol{x}^{(i)}_{t-1|t-1}}_t\right) \\ \boldsymbol{a}^{(i)}_{t-1} + \boldsymbol{v}^{(i),\boldsymbol{a}^{(i)}_{t-1}}_t \end{pmatrix},\end{aligned} \tag{6}$$

where $\boldsymbol{v}^{(i),\boldsymbol{x}^{(i)}_{t-1|t-1}}_t$, $\boldsymbol{v}^{(i),\boldsymbol{a}^{(i)}_{t-1}}_t$ are noises applied to the state and the parameter, respectively.

B. Problem of PF

PF has a serious problem in that the estimation accuracy deteriorates when PF estimates not only states but also parameters of state space models. The reason is as follows. The estimation accuracy of PF deteriorates if the diversity of particles in an ensemble is lost. The diversity of particles tends to lost when the particles include parameters because the parameters hardly change with time according to the state equation, (6), where the system noise $\boldsymbol{v}^{(i),\boldsymbol{a}^{(i)}_t}_t$ is usually set to small because the parameters are constant.

IV. The Proposed Method

A. Basic Ideas

In this paper, we propose a method combining PF and natural evolution strategy (NES) [9] in order to overcome the problem of PF which is that the performance is degraded when parameters are estimated together with states. As shown in Fig. 1, the proposed method performs state estimation and parameter estimation by PF and NES, respectively. In parameter estimation, the proposed method employs the normal distribution as a probability distribution of parameter $\boldsymbol{a}$ and updates the probability distribution by NES. Assume that a filtering ensemble $\{\boldsymbol{x}^{(i)}_{t-1|t-1}\}^N_{i=1}$ and a probability distribution of parameter $p(\boldsymbol{a}|\boldsymbol{\mu}_{t-1}, D_{t-1})$ at time $t-1$ as shown in Fig. 1a are given. The proposed method, first, generates n parameters $\{\boldsymbol{a}^{(i)}_t\}^n_{i=1}$ according to the probability distribution of parameter $p(\boldsymbol{a}|\boldsymbol{\mu}_{t-1}, D_{t-1})$ which is a normal distribution, as shown in Fig. 1b, and makes each of them correspond to the estimation of the true state $\hat{\boldsymbol{x}}_{t-1|t-1}$ calculated from the filtering ensemble at the time $t-1$, $\{(\hat{\boldsymbol{x}}_{t-1|t-1}, \boldsymbol{a}^{(i)}_t)\}^n_{i=1}$, where n is the number of individuals generated by NES and $\boldsymbol{\mu}_{t-1}$ and D_{t-1} are the mean vector and the covariance matrix of the probability distribution of parameter at time $t-1$, respectively. In this paper, the estimation of the true state is the mean of the filtering ensemble $\hat{\boldsymbol{x}}_{t-1|t-1} = \frac{1}{N}\sum^N_{i=1} \boldsymbol{x}^{(i)}_{t-1|t-1}$ [13]. Next, as shown in Fig. 1c, the proposed method calculates a prediction of the estimation of the true state $\hat{\boldsymbol{x}}^{\boldsymbol{a}^{(i)}_t}_{t|t-1} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{t-1|t-1}, \boldsymbol{a}^{(i)}_t, \boldsymbol{v}^{(i)}_t)$ using $\boldsymbol{a}^{(i)}_t$ as a parameter. Next, as shown in Fig. 1d, the proposed method calculates the likelihood $\lambda(\hat{\boldsymbol{x}}^{\boldsymbol{a}^{(i)}_t}_{t|t-1})$ of particle $\hat{\boldsymbol{x}}^{\boldsymbol{a}^{(i)}_t}_{t|t-1}$ based on the observation to calculate the evaluation value $h(\boldsymbol{a}^{(i)}_t)$ of parameters $\boldsymbol{a}^{(i)}_t$. The evaluation value $h(\boldsymbol{a})$ is set to $h(\boldsymbol{a}^{(i)}_t) = -\lambda(\hat{\boldsymbol{x}}^{\boldsymbol{a}^{(i)}_t}_{t|t-1})$ by regarding an individual having a high likelihood as a good individual. The minus on the likelihood is to make the problem a minimization one. Next, as shown in Fig. 1e, the proposed method updates the probability distribution of parameter using $\{(\boldsymbol{a}^{(i)}_t, h(\boldsymbol{a}^{(i)}_t))\}^n_{i=1}$ in order for parameters having higher evaluation values to be more likely to be generated from the probability distribution of parameter at the next time step. As a result, the proposed method obtains the probability distribution of parameter $p(\boldsymbol{a} \mid \boldsymbol{\mu}_t, D_t)$. A specific method of updating the probability distribution of parameter will be described in Section IV-B. Finally, as shown in Fig. 1f, the proposed method calculates the filtering ensemble at time t, $\{\boldsymbol{x}^{(i)}_{t|t}\}^N_{i=1}$, from the one at $t-1$, $\{\boldsymbol{x}^{(i)}_{t-1|t-1}\}^N_{i=1}$. To do so, the proposed method, first, performs the prediction step of PF using $\boldsymbol{\mu}_t$ as the parameter of the state equation to obtain a prediction ensemble $\{\boldsymbol{x}^{(i)}_{t|t-1} = \boldsymbol{f}(\boldsymbol{x}^{(i)}_{t-1|t-1}, \boldsymbol{\mu}_t, \boldsymbol{v}^{(i)}_t)\}^N_{i=1}$. Then, the proposed method calculates the likelihood of each particle in the prediction ensemble $\{\boldsymbol{x}^{(i)}_{t|t-1}\}^N_{i=1}$ and performs the filtering step of PF to obtain a filtering ensemble $\{\boldsymbol{x}^{(i)}_{t|t}\}^N_{i=1}$ at time t. Finally, the proposed method calculates and outputs the estimation of the true state $\hat{\boldsymbol{x}}_{t|t} = \frac{1}{N}\sum^N_{i=1} \boldsymbol{x}^{(i)}_{t|t}$.

The proposed method repeats the above steps shown in Fig. 1a to Fig. 1f until time T.

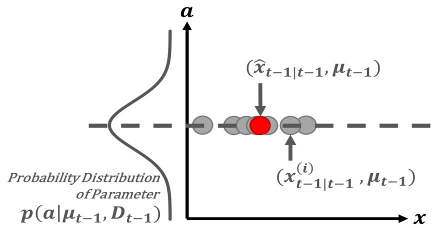
B. Updating of Probability Distribution of Parameter by Natural Evolution Strategy

The proposed method updates the probability distribution of parameter $p(\boldsymbol{a} \mid \boldsymbol{\mu}, D)$ using the Separable Natural Evolution Strategy (SNES) [15].

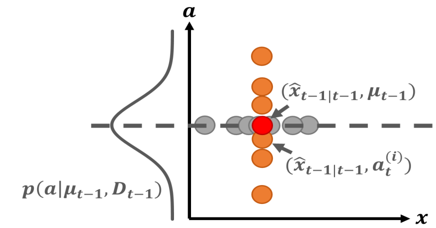
NES is one of the most powerful black-box function optimization methods. Instead of directly finding the optimal solution $\boldsymbol{a}^*$ which minimizes the evaluation value of an objective function $h(\boldsymbol{a})$, NES searches for the optimal hyperparameter $\boldsymbol{\theta}^*$ of a probability distribution $p(\boldsymbol{a} \mid \boldsymbol{\theta})$ which minimizes the expected evaluation value $J(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{\theta}}[h(\boldsymbol{a})] = \int h(\boldsymbol{a})p(\boldsymbol{a} \mid \boldsymbol{\theta})d\boldsymbol{a}$ by using the natural gradient method [11]. The update formula of $\boldsymbol{\theta}$ by NES is as follows.

$$\boldsymbol{\theta}_t \leftarrow \boldsymbol{\theta}_{t-1} - \eta F^{-1}(\boldsymbol{\theta}_{t-1}) \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_{t-1}), \tag{7}$$

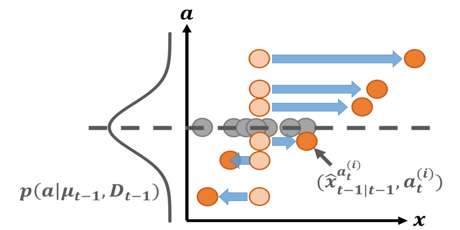
where t is an iteration number, $\eta(>0)$ is a learning rate, $F^{-1}(\boldsymbol{\theta}_{t-1})$ is the inverse matrix of the Fisher information matrix and $F^{-1}(\boldsymbol{\theta}_{t-1})\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_{t-1})$ is the natural gradient. $\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$ can be approximated as $\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \simeq$



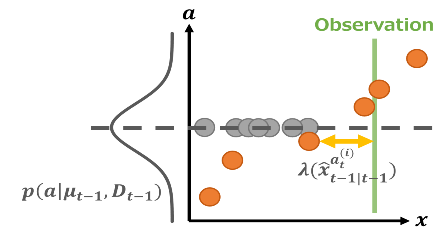
(a) A filtering ensemble and a probability distribution of parameter at time $t-1$. $\hat{\boldsymbol{x}}_{t-1|t-1} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}^{(i)}_{t-1|t-1}$ represents the estimation of the true state at time $t-1$.



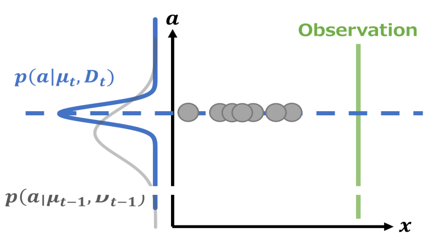
(b) The proposed method generates n parameters $\{\boldsymbol{a}^{(i)}_t\}_{i=1}^{n}$ according to the probability distribution of parameter $p(\boldsymbol{a}|\boldsymbol{\mu}_{t-1}, D_{t-1})$ and makes each of them correspond to $\hat{\boldsymbol{x}}_{t-1|t-1}$.



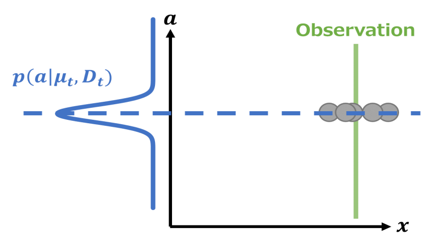
(c) The proposed method calculates a prediction $\hat{\boldsymbol{x}}^{\boldsymbol{a}^{(i)}_t}_{t|t-1} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{t-1|t-1}, \boldsymbol{a}^{(i)}_t, \boldsymbol{v}^{(i)}_t)$ using parameter $\boldsymbol{a}^{(i)}_t$.



(d) Based on the observation, the proposed method calculates the likelihood $\lambda(\hat{\boldsymbol{x}}^{\boldsymbol{a}^{(i)}_t}_{t|t-1})$ of $\hat{\boldsymbol{x}}^{\boldsymbol{a}^{(i)}_t}_{t|t-1}$ and the evaluation value $h(\boldsymbol{a}^{(i)}_t)$ of parameter $\boldsymbol{a}^{(i)}_t$.



(e) The proposed method updates the probability distribution of parameter p by NES using $\{\boldsymbol{a}^{(i)}_t, h(\boldsymbol{a}^{(i)}_t)\}_{i=1}^{n}$.



(f) The proposed method calculates the filtering ensemble at time t, $\{\boldsymbol{x}^{(i)}_{t|t}\}_{i=1}^{N}$, from the one at $t-1$, $\{\boldsymbol{x}^{(i)}_{t-1|t-1}\}_{i=1}^{N}$, by performing the prediction step of PF using $\boldsymbol{\mu}_t$ as the parameter of the state equation and the filtering step of PF using the likelihood based on the observation.

Fig. 1. An overview of the proposed method. The horizontal and vertical axes represent the state and the parameter, respectively. The probability distribution to the left of the vertical axis represents the probability distribution of parameter used by NES to generate offspring parameters. The yellow-green vertical line represents the observation.

$\frac{1}{n}\sum_{i=1}^{n} h(\boldsymbol{a}^{(i)})\nabla_{\boldsymbol{\theta}} \ln p(\boldsymbol{a}^{(i)} \mid \boldsymbol{\theta})$ using n solutions $\{\boldsymbol{a}^{(i)}\}_{i=1}^{n}$ generated according to $p(\boldsymbol{a} \mid \boldsymbol{\theta})$.

SNES uses a multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}, D)$ as a probability distribution p where the covariance matrix D is a diagonal matrix in order to improve the convergence speed in variable-separable objective functions. SNES updates the mean vector $\boldsymbol{\mu}$ and the covariance matrix D of the multivariate normal distribution as follows.

$$\boldsymbol{\mu}_t = \boldsymbol{\mu}_{t-1} + \eta_\mu \cdot D^{\frac{1}{2}}_{t-1}\nabla_{\boldsymbol{\mu}} J, \tag{8}$$

$$D^{\frac{1}{2}}_t = D^{\frac{1}{2}}_{t-1}\exp(\eta_D/2 \cdot \nabla_D J). \tag{9}$$

where η_μ and η_D are learning rates and $\nabla_{\boldsymbol{\mu}} J$ and $\nabla_D J$ are the gradients given by

$$\nabla_{\boldsymbol{\mu}} J = \sum_{i=1}^{n} u^{(i)} \cdot \boldsymbol{s}^{(i)}, \tag{10}$$

$$\nabla_D J = \sum_{i=1}^{n} u^{(i)} \cdot \mathrm{diag}(\boldsymbol{s}^{(i:n)} \odot \boldsymbol{s}^{(i:n)} - 1), \tag{11}$$

$$\boldsymbol{s}^{(i)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), \tag{12}$$

where $\odot$ is the element product and $\boldsymbol{s}^{(i:N)}$ is the ith solution when $\boldsymbol{a}^{(i)}_t = \boldsymbol{\mu}_{t-1} + D^{\frac{1}{2}}_{t-1}\boldsymbol{s}^{(i)}$ are arranged in the descending order of evaluation value $h(\boldsymbol{a}^{(i)}_t)$. In this paper, we use a linear weight instead of the weight using log used in SNES [15] so as not to trust too much a good solution in consideration of the existence of noise. Specifically, letting $u^{(i)}$ be the weight of the i-th solution,

$$u^{(i)} = \frac{1.0 - \frac{i}{n}}{\sum_{j=1}^{n}\left(1.0 - \frac{j}{n}\right)} - \frac{1}{n}. \tag{13}$$

The recommended value of η_D is given by $\eta_D = \dfrac{3 + log(d_{\mathrm{p}})}{5\sqrt{d_{\mathrm{p}}}}$ [15], where d_{p} is the dimension of the solution $\boldsymbol{a}$.

C. Algorithm

Algorithm 2 shows the algorithm of the proposed method. In this section, only the difference from PF shown in Algorithm 1 is explained. In line 2 to 5, the proposed method generates parameters $\boldsymbol{a}_t^{(i)}$ according to the probability distribution of parameter $p(\boldsymbol{a} \mid \boldsymbol{\mu}_{t-1}, D_{t-1})$. In line 3, vector $\boldsymbol{s}^{(i)}$ is generated according to the standard normal distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. In line 4, parameter $\boldsymbol{a}_t^{(i)}$ is calculated from $\boldsymbol{s}^{(i)}$. In line 6, the proposed method gets an observation $\boldsymbol{y}_t$. In line 7 to 12, the proposed method calculates the evaluation value $h(\boldsymbol{a}_t^{(i)})$ of each parameter $\boldsymbol{a}_t^{(i)}$. In line 8, the system noise $\boldsymbol{v}_t^{(i)}$ is generated. In line 9, $\hat{\boldsymbol{x}}_{t|t-1}^{\boldsymbol{a}_t^{(i)}}$ is calculated as the prediction of $\hat{\boldsymbol{x}}_{t-1|t-1}$ using each parameter $\boldsymbol{a}_t^{(i)}$. In line 10, the likelihood $\lambda(\hat{\boldsymbol{x}}_{t|t-1}^{\boldsymbol{a}_t^{(i)}})$ of the prediction $\hat{\boldsymbol{x}}_{t|t-1}^{\boldsymbol{a}_t^{(i)}}$ is calculated. In line 11, the evaluation value $h(\boldsymbol{a}_t^{(i)})$ of each parameter $\boldsymbol{a}_t^{(i)}$ is calculated. In line 13 to 16, the proposed method updates the probability distribution of parameter p. In line 13 and 14, the gradients of the mean vector and the covariance matrix of the probability distribution of parameter are calculated. In line 15 and 16, the mean vector and the covariance matrix are updated. In line 17 to 21, the proposed method calculates the prediction ensemble $\{\boldsymbol{x}_{t|t-1}^{(i)}\}_{i=1}^{N}$ and the likelihood $\lambda(\boldsymbol{x}_{t|t-1}^{(i)})$ of each prediction particle $\boldsymbol{x}_{t|t-1}^{(i)}$. In line 19, the prediction ensemble $\{\boldsymbol{x}_{t|t-1}^{(i)}\}_{i=1}^{N}$ at time t is calculated from the filtering ensemble $\{\boldsymbol{x}_{t-1|t-1}^{(i)}\}_{i=1}^{N}$ at time $t-1$ using the mean vector $\boldsymbol{\mu}_t$ of the updated probability distribution of parameter as the parameter of the state equation.

V. Experiments

In this section, we conduct some experiments in order to confirm that the proposed method shows better estimation performance than the original PF in problems where state and parameter are estimated simultaneously. We use two problems of estimating the states and the parameters of the Van der Pol model [16] and the Lorenz model [17].

A. Van der Pol Model

In this problem, the state $\boldsymbol{x} = (x_1, x_2)^{\mathrm{T}}$ and the parameter $\boldsymbol{a} = (a_1, a_2, a_3, a_4)^{\mathrm{T}}$ of the Van der Pol model are estimated.

Algorithm 2 The Proposed Method

Require: $p_{\mathrm{s}}(\cdot), p_{\mathrm{v}}(\cdot), \boldsymbol{f}(\cdot), p(\cdot), N, n, \boldsymbol{\mu}_0, D_0, \eta_{\boldsymbol{\mu}}, \eta_D, T$
Input: $\{\boldsymbol{x}_{0|0}^{(i)}\}_{i=1}^{N}$
Output: $\hat{\boldsymbol{x}}_{t|t}\ (t = 1, \cdots, T)$
1: **for** $t = 1, \cdots, T$ **do**
2:   **for** $i = 1, \cdots, n$ **do**
3:     Draw sample $\boldsymbol{s}^{(i)} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
4:     $\boldsymbol{a}_t^{(i)} = \boldsymbol{\mu}_{t-1} + D_{t-1}^{\frac{1}{2}} \boldsymbol{s}^{(i)}$
5:   **end for**
6:   Observe $\boldsymbol{y}_t$
7:   **for** $i = 1, \cdots, n$ **do**
8:     Generate $\boldsymbol{v}_t^{(i)} \sim p_{\mathrm{s}}(\boldsymbol{v})$
9:     $\hat{\boldsymbol{x}}_{t|t-1}^{\boldsymbol{a}_t^{(i)}} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{t-1|t-1}, \boldsymbol{a}_t^{(i)}, \boldsymbol{v}_t^{(i)})$
10:     Calculate $\lambda(\hat{\boldsymbol{x}}_{t|t-1}^{\boldsymbol{a}_t^{(i)}})$ using $\boldsymbol{y}_t$
11:     Calculate $h(\boldsymbol{a}_t^{(i)})$
12:   **end for**
13:   $\nabla_{\boldsymbol{\mu}} J = \Sigma_{i=1}^{n} u^{(i)} \cdot \boldsymbol{s}^{(i:n)}$
14:   $\nabla_D J = \Sigma_{i=1}^{n} u^{(i)} \cdot \mathrm{diag}(\boldsymbol{s}^{(i:n)} \odot \boldsymbol{s}^{(i:n)} - 1)$
15:   $\boldsymbol{\mu}_t = \boldsymbol{\mu}_{t-1} + \eta_{\boldsymbol{\mu}} \cdot D_{t-1}^{\frac{1}{2}} \nabla_{\boldsymbol{\mu}} J$
16:   $D_t^{\frac{1}{2}} = D_{t-1}^{\frac{1}{2}} \cdot \exp(\eta_D/2 \cdot \nabla_D J)$
17:   **for** $i = 1, \cdots, N$ **do**
18:     Generate $\boldsymbol{v}_t^{(i)} \sim p_{\mathrm{s}}(\boldsymbol{v})$
19:     $\boldsymbol{x}_{t|t-1}^{(i)} = \boldsymbol{f}(\boldsymbol{x}_{t-1|t-1}^{(i)}, \boldsymbol{\mu}_t, \boldsymbol{v}_t^{(i)})$
20:     Calculate $\lambda(\boldsymbol{x}_{t|t-1}^{(i)})$ using $\boldsymbol{y}_t$
21:   **end for**
22:   **for** $i = 1, \cdots, N$ **do**
23:     Choose a particle $\boldsymbol{x}_{t|t}^{(i)}$ from $\{\boldsymbol{x}_{t|t-1}^{(k)}\}_{k=1}^{N}$ with replacement with probabilities propotional to $\{\lambda(\boldsymbol{x}_{t|t-1}^{(k)})\}_{k=1}^{N}$.
24:   **end for**
25:   $\hat{\boldsymbol{x}}_{t|t} = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{x}_{t|t}^{(i)}$
26: **end for**

The state equation for the Van der Pol model is given by (14) and (15) and the observation equation is given by (16).

$$\begin{aligned}\boldsymbol{x}_t &= \boldsymbol{f}\left(\boldsymbol{x}_{t-1}, \boldsymbol{a}_{t-1}, \boldsymbol{v}_t\right) \\ &= \boldsymbol{x}_{t-1} + \Delta T \boldsymbol{g}\left(\boldsymbol{x}_{t-1}, \boldsymbol{a}_{t-1}\right) + \sqrt{\Delta T}\boldsymbol{v}_t, \end{aligned} \tag{14}$$

$$\boldsymbol{g}\left(\boldsymbol{x}, \boldsymbol{a}\right) = \begin{pmatrix} a_1 x_2 \\ a_2 x_2 - a_3 x_1^2 x_2 - a_4 x_1 \end{pmatrix}, \tag{15}$$

$$\boldsymbol{y}_t = \boldsymbol{x}_t + \boldsymbol{w}_t, \tag{16}$$

where $\boldsymbol{x} = (x_1, x_2)^{\mathrm{T}}$ is a state, $\boldsymbol{a} = (a_1, a_2, a_3, a_4)^{\mathrm{T}}$ is a parameter and ΔT is the step size of the discretization for the Euler method. In this experiment, the initial true state is $\boldsymbol{x}_0 = (0.2, 0.1)^{\mathrm{T}}$, the true parameter is $\boldsymbol{a}^* = (1.0, 1.0, 1.0, 1.0)^{\mathrm{T}}$ and $\Delta T = 10^{-1}$.

B. Lorenz Model

In this problem, the state $\boldsymbol{x} = (x_1, x_2, x_3)^{\mathrm{T}}$ and the parameter $\boldsymbol{a} = (a_1, a_2, a_3)^{\mathrm{T}}$ of the Lorenz model are estimated. The state equation for the Van der Pol model is given by (17) and (18) and the observation equation is given by (19).

$$\begin{aligned}\boldsymbol{x}_t &= \boldsymbol{f}(\boldsymbol{x}_{t-1}, \boldsymbol{a}_{t-1}, \boldsymbol{v}_t) \\ &= \boldsymbol{x}_{t-1} + \Delta T \boldsymbol{g}(\boldsymbol{x}_{t-1}, \boldsymbol{a}_{t-1}) + \sqrt{\Delta T}\boldsymbol{v}_t,\end{aligned} \tag{17}$$

$$\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{a}) = \begin{pmatrix} -a_1(x_1 - x_2) \\ -x_1 x_3 + a_2 x_1 - x_2 \\ x_1 x_2 - a_3 x_3 \end{pmatrix}, \tag{18}$$

$$\boldsymbol{y}_t = \boldsymbol{x}_t + \boldsymbol{w}_t, \tag{19}$$

where $\boldsymbol{x} = (x_1, x_2, x_3)^{\mathrm{T}}$ is a state, $\boldsymbol{a} = (a_1, a_2, a_3)^{\mathrm{T}}$ is a parameter and ΔT is the step size of the discretization for the Euler method. In this experiment, the initial true state is $\boldsymbol{x}_0 = (-16.0, -21.6, 34.2)^{\mathrm{T}}$, the true parameters is $\boldsymbol{a}^* = (10, 28, \frac{8}{3})^{\mathrm{T}}$ and $\Delta T = 10^{-2}$.

C. Experimental Method

The experimental method is the identical-twin experiment [18]. The experiment is performed in the following procedure.

1) Give an initial true state $\boldsymbol{x}_0^*$ and a true parameter $\boldsymbol{a}^*$.
2) Generate a true state sequence $\{\boldsymbol{x}_0^*, \boldsymbol{x}_1^*, \cdots, \boldsymbol{x}_T^*\}$ and an observation sequence $\{\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_T\}$ by the state equation and the observation equation.
3) Generate a filtering ensemble $\{\boldsymbol{x}_{0|0}^{(i)}\}_{i=1}^N$ at time $t = 0$ from the initial ensemble generation distribution $p_{\mathrm{init}}(\boldsymbol{x})$ and perform state estimation from $t = 1$ to $t = T$.
4) Perform 100 trials changing random seeds, where a trial is to perform step 3.

D. Evaluation criterion

The mean square error (MSE) in successful trials is used as a criterion. A successful trial is a trial in which the state does not diverge during the trial. If the norm of the center of gravity of the ensemble exceeds the threshold value $l_{\mathrm{threshold}}$, it is determined that the state diverges. The smaller MSE is, the better the performance is.

MSE is the average of the square errors between the true state $\boldsymbol{x}_t^*$ and its estimation $\hat{\boldsymbol{x}}_{t|t}$ from the start time to the end time. MSE is given by (20).

$$\mathrm{MSE} = \frac{1}{T}\sum_{t=1}^{T}\left(\boldsymbol{x}_t^* - \hat{\boldsymbol{x}}_{t|t}\right)^2. \tag{20}$$

The estimation of the true state is the mean of the filtering ensemble $\hat{\boldsymbol{x}}_{t|t} = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{x}_{t|t}^{(i)}$.

E. Settings

1) Common: The common settings are as follows. Since, in most real-world applications, calculating the state equation is the most expensive calculation in PF and the proposed method, the numbers of calculations of the state equation for prediction in the proposed method and PF should be the same, which means that $N_{\mathrm{PF}} = n + N_{\mathrm{proposed}}$, where N_{PF} is N in PF and N_{proposed} is N in the proposed method. The final time is $T = 20,000$. The threshold for determining whether the state diverges or not is $l_{\mathrm{threshold}} = 1.0 \times 10^5$. The probability density functions of the system noise $\boldsymbol{v}_t$ and the observation noise $\boldsymbol{\omega}_t$ are $p_{\mathrm{s}}(\boldsymbol{v}) = \mathcal{N}\left(\boldsymbol{0}, \sqrt{0.01}^2\boldsymbol{I}\right)$ and $p_{\mathrm{m}}(w) = \mathcal{N}\left(\boldsymbol{0}, \sqrt{0.01}^2\boldsymbol{I}\right)$, respectively. The likelihood function is $\lambda(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{y} - \boldsymbol{x}, \sqrt{0.01}^2\boldsymbol{I})$. Since there is not the recommended value for $\eta_{\boldsymbol{\mu}}$, $\eta_{\boldsymbol{\mu}} = 0.1$ is used in this paper.

2) Experimenting Van der Pol Model: N_{PF}, N_{proposed} and n are set to 80, 50 and 30, respectively. The states in the initial ensemble are generated according to the normal distribution $p_{\mathrm{init}}(\boldsymbol{x}) = \mathcal{N}\left((0.2, 0.1)^{\mathrm{T}}, \sqrt{0.5}^2\boldsymbol{I}\right)$. The system noise of the parameter $\boldsymbol{a} = (a_1, a_2, a_3, a_4)^{\mathrm{T}}$ in PF follows $\mathcal{N}(0, \sqrt{10^{-5}}^2\boldsymbol{I})$. $\boldsymbol{a}$ in the initial ensemble is generated according to $\mathcal{N}\left(\boldsymbol{0}, \sqrt{2.0}^2\boldsymbol{I}\right)$ in PF. The initial values of $\boldsymbol{\mu}$ and D_0 of the probability distribution of parameter $p(\boldsymbol{a}|\boldsymbol{\mu}, D)$ in the proposed method are $\boldsymbol{\mu}_0 = \boldsymbol{0}$ and $D = 2.0\boldsymbol{I}$, respectively.

3) Experimenting Lorenz Model: N_{PF}, N_{proposed} and n are set to 400, 200 and 200, respectively. The states in the initial ensemble are generated according to the normal distribution $p_{\mathrm{init}}(\boldsymbol{x}) = \mathcal{N}\left((-16.0, -21.6, 34.2)^{\mathrm{T}}, \sqrt{1.0}^2\boldsymbol{I}\right)$. The system noise of the parameter $\boldsymbol{a} = (a_1, a_2, a_3)$ in PF follows $\mathcal{N}(0, \sqrt{10^{-5}}^2\boldsymbol{I})$. $\boldsymbol{a}$ in the initial ensemble is generated according to $\mathcal{N}\left((10.0 + 0.5, 28.0 + 0.5, \frac{8}{3} + 0.5)^{\mathrm{T}}, \sqrt{1.0}^2\boldsymbol{I}\right)$. The initial values of $\boldsymbol{\mu}$ and D of the probability distribution of parameter $p(\boldsymbol{a}|\boldsymbol{\mu}, D)$ in the proposed method are $\boldsymbol{\mu}_0 = (10.0 + 0.5, 28.0 + 0.5, \frac{8}{3} + 0.5)^{\mathrm{T}}$ and $D_0 = 1.0\boldsymbol{I}$, respectively.

F. Results

Fig. 2 and Fig. 3 show the box plots of the MSE values of the states and the parameters in successful trials out of 100 trials in the experiments using the Van der Pol model and the Lorenz model, respectively. From Fig. 2 and Fig. 3, it is confirmed that the proposed method was superior to PF in terms of MSE. The numbers of successful trials of the proposed method and PF in the experiments using the Van der Pol model were 100 and 57, respectively, and those in the experiments using the Lorenz model were both 100. From the above two experiments, it can be confirmed that the proposed method achieves better estimation performance than the existing method, PF.

VI. Discussion

Fig. 4 shows the time transition of the states and that of the parameters estimated by the proposed method and PF in the median trial of the experiment using the Van der Pol model. Fig. 5 shows the time transition of the states and that of the parameters estimated by the proposed method and PF in the median trial of the experiment using the Lorenz model. The median trial is the trial with the better evaluation value between the trials with the closest values from the median. From Fig. 4b and Fig. 5b, in PF, the diversity of particles was lost as early as around time $t = 5$ and the estimated parameter converged at the position far from the true parameter. As a result, the estimation accuracy of the PF was degraded. On

(a) MSE of states (b) MSE of parameters

Fig. 2. The box plots of MSE values of the states and the parameters in the successful trials out of 100 trials of the proposed method and PF in the experiment using the Van der Pol model. The smaller MSE is, the better performance is. The numbers of success trials of the proposed method and PF are 100 and 57, respectively. The range between the bottom of the whisker and that of the box represents the best 25% trials. The range between the bottom of the box and the orange line represents the best 25% to 50% trials. The range between the orange line and the top of the box represents the best 50% to 75% trial. The range between the top of the box and the top of the whisker represents the best 75% to 100% trials. The points represent outliers.

(a) MSE of states (b) MSE of parameters

Fig. 3. The box plots of MSE values of the states and the parameters in the successful trials out of 100 trials of the proposed method and PF in the experiment using the Lorenz model. The smaller MSE is, the better performance is. The numbers of success trials of the proposed method and PF are both 100. The range between the bottom of the whisker and that of the box represents the best 25% trials. The range between the bottom of the box and the orange line represents the best 25% to 50% trials. The range between the orange line and the top of the box represents the best 50% to 75% trial. The range between the top of the box and the top of the whisker represents the best 75% to 100% trials. The points represent outliers.

the other hand, the proposed method succeeded in finding a parameter closer to the true parameter than PF.

VII. Conclusion

In this paper, we proposed a new sequential estimation method that overcomes a problem of the particle filter (PF) that is one of the most powerful sequential estimation methods and demonstrated that the proposed method showed better performance than PF in terms of estimation accuracy on two benchmark problems. The problem of PF is that the estimation accuracy deteriorates when PF estimates parameters in addition to states simultaneously. The proposed method estimates states by PF and parameters by the separable natural evolution strategy (SNES). In order to show the effectiveness of the proposed method, we compared the performance of the proposed method and that of PF on two problems. One is to estimate the state whose dimension is two and the parameter whose dimension is four of the Van der Pol model. The other is to estimate the state whose dimension is three and the parameter whose dimension is three of the Lorenz model. In the Van der Pol model, the median MSE values of the state and the parameter of the proposed method were 0.003610 and 0.01468 and those of PF were 4.228 and 6.520, respectively. In the Lorenz model, the median MSE values of the state and the parameter of the proposed method were 0.002639 and 0.003479 and those of PF were 309.5 and 1.470, respectively.

As future work, we would like to propose a new method that employs SNES for estimating parameters and the ensemble Kalman filter (EnKF) [6] for estimating states because EnKF can be applied to higher-dimensional problems than PF. In addition, we would like to consider to use xNES [19] that employs a normal distribution with a full-rank covariance matrix instead of SNES. Furthermore, we would like to apply the proposed method to some real-world estimation problems.

References

[1] R. Xiong, Y. Zhang, H. He, X. Zhou, and M. G. Pecht, “A double-scale, particle-filtering, energy state prediction algorithm for lithium-ion batteries,” *IEEE Transactions on Industrial Electronics*, vol. 65, no. 2, pp. 1526–1538, 2018.

[2] G. Yamauchi, D. Suzuki, and K. Nagatani, “Online slip parameter estimation for tracked vehicle odometry on loose slope,” in *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2016, pp. 227–232.

[3] A. Botto, E. Belluco, and M. Camporese, “Multi-source data assimilation for physically based hydrological modeling of an experimental hillslope,” *Hydrology and Earth System Sciences*, vol. 22, no. 8, pp. 4251–4266, 2018.

[4] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Journal of basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

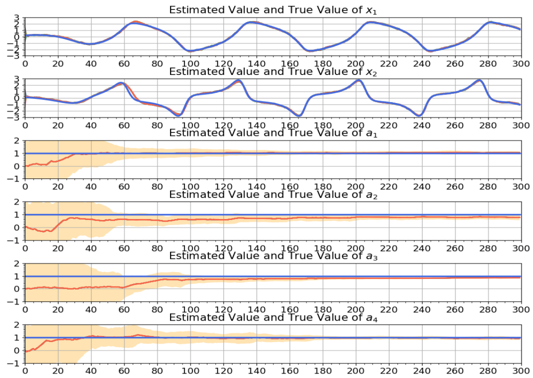
[5] Sorenson and H. Wayne, *Kalman filtering: theory and application*. IEEE, 1985.

[6] G. Evensen, “Sequential data assimilation with a nonlinear quasi - geostrophic model using Monte Carlo methods to forecast error statistics,” *Journal of Geophysical Research: Oceans*, vol. 99, pp. 10 143–10 162, 1994.

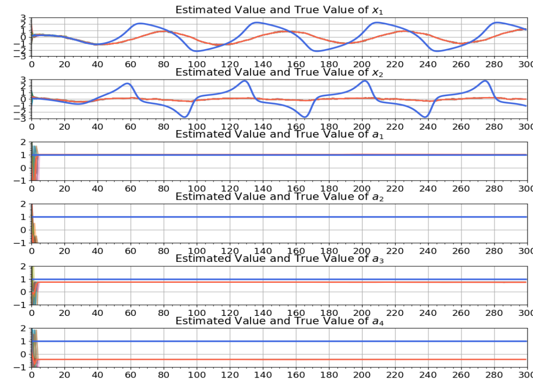
[7] S. J. Julier and J. K. Uhlmann, “New extension of the Kalman filter to nonlinear systems,” *Proc. of SPIE*, pp. 182–193, 1997.

[8] N. Gordon, D. Salmond, and A. Smith, “Novel approach to nonlinear/non-Gaussian Bayesian state estimation,” *IEE Proceedings F - Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, 1993.

[9] D. Wierstra, T. Schaul, J. Peters, and J. Schmidhuber, “Natural evolution strategies,” in *2008 IEEE Congress on Evolutionary Computation (IEEE*

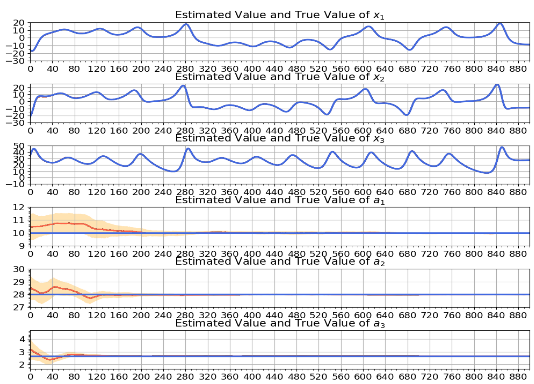


(a) Time transition of x_1 and x_2 (upper 2 diagrams) and a_1, a_2, a_3 and a_4 (lower 4 diagrams) estimated by the proposed method.

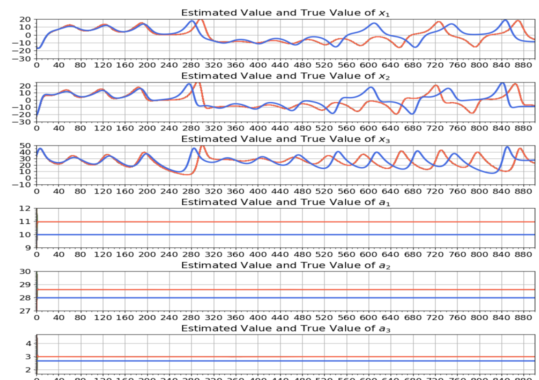


(b) Time transition of x_1 and x_2 (upper 2 diagrams) and a_1, a_2, a_3 and a_4 (lower 4 diagrams) estimated by PF.

Fig. 4. The time transition diagrams of the estimated state, the estimated parameter and the ensemble by the proposed method and PF in the experiment using Van der Pol model. The time transition diagrams of x_1 and x_2 of the state and a_1, a_2, a_3 and a_4 of the parameter in (15) are shown in order from the top. The horizontal axis represents time and the vertical one represents the value of each component of the state or the parameter. The blue lines represent the true values and the red lines represent the values estimated by the proposed method or PF. The lines with other colors represent the time transitions of each ensemble particle. The yellow area in Fig. 4a represents the standard deviation of the probability distribution of parameter in the proposed method.



(a) Time transition of x_1, x_2 and x_3 (upper 3 diagrams) and a_1, a_2 and a_3 (lower 3 diagrams) estimated by the proposed method.



(b) Time transition of x_1, x_2 and x_3 (upper 3 diagrams) and a_1, a_2 and a_3 (lower 3 diagrams) estimated by PF

Fig. 5. The time transition diagrams of the estimated state, the estimated parameter and the ensemble by the proposed method and PF in the experiment using Lorenz model. The time transition diagrams of x_1, x_2 and x_3 of the state and a_1, a_2 and a_3 of the parameter in (18) are shown in order from the top. The horizontal axis represents time and the vertical one represents the value of each component of the state or the parameter. The blue lines represent the true values and the red lines represent the values estimated by the proposed method or PF. The lines with other colors represent the time transitions of each ensemble particle. The yellow area in Fig. 5a represents the standard deviation of the probability distribution of parameter in the proposed method.

World Congress on Computational Intelligence). IEEE, 2008, pp. 3381–3387.

[10] D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, J. Peters, and J. Schmidhuber, "Natural evolution strategies," *Journal of Machine Learning Research*, vol. 15, pp. 949–980, 2014.

[11] S. Amari, "Natural gradient works efficiently in learning," *Neural Computation*, vol. 10, no. 2, pp. 251–276, 1998.

[12] G. Kitagawa, "Monte carlo filter and smoother for non-gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1–25, 1996.

[13] Li Liang-qun, Ji Hong-bing, and Luo Jun-hui, "The iterated extended kalman particle filter," in *IEEE International Symposium on Communications and Information Technology, 2005. ISCIT 2005.*, vol. 2, Oct 2005, pp. 1213–1216.

[14] G. Kitagawa, "A self-organizing state-space model," *Journal of the American Statistical Association*, pp. 1203–1215, 1998.

[15] T. Schaul, "Benchmarking exponential natural evolution strategies on the noiseless and noisy black-box optimization testbeds," in *Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation*. ACM, 2012, pp. 213–220.

[16] B. Van der Pol, "The nonlinear theory of electric oscillations," *Proceedings of the Institute of Radio Engineers*, vol. 22, no. 9, pp. 1051–1086, 1934.

[17] E. N. Lorenz, "Deterministic nonperiodic flow," *Journal of the atmospheric sciences*, vol. 20, no. 2, pp. 130–141, 1963.

[18] J. Kao, D. Flicker, R. Henninger, S. Frey, M. Ghil, and K. Ide, "Data assimilation with an extended kalman filter for impact-produced shock-wave dynamics," *Journal of Computational Physics*, vol. 196, no. 2, pp. 705–723, 2004.

[19] T. Glasmachers, T. Schaul, S. Yi, D. Wierstra, and J. Schmidhuber, "Exponential natural evolution strategies," in *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, 2010, pp. 393–400.