Simultaneously estimating and controlling nonlinear neuronal dynamics based on sequential Monte Carlo framework

Omi, Taketo

Omori, Toshiaki

Paper

Simultaneously estimating and controlling nonlinear neuronal dynamics based on sequential Monte Carlo framework

Taketo Omi [iD] **¹ and Toshiaki Omori** [iD] **¹ ᵃ⁾**

¹ *Department of Electrical and Electronic Engineering, Graduate School of Engineering, Kobe University 1-1 Rokkodai-cho, Nada-ku, Kobe 657-8501, Japan*

ᵃ⁾ *omori@eedept.kobe-u.ac.jp*

Abstract: Estimating and controlling nonlinear neuronal system are crucial for understanding the neuronal dynamics and brain functions. However, it is challenging to control the nonlinear system including unobservable state and unknown dynamics. We propose a framework for estimating and controlling an individual neuron by leveraging the sequential Monte Carlo method (SMC). We derive an online algorithm based on the expectation-maximization algorithm and constitute the control law by employing the SMC-based model predictive control. We verify the effectiveness of the proposed method using simulation environments. The results suggest we can simultaneously estimate the latent variables and the parameters and control neuronal state.

Key Words: nonlinear dynamics, neuronal dynamics, statistical machine learning, data-driven science, neuroscience

1. Introduction

Estimating and controlling nonlinear neuronal system are crucial for understanding the neuronal system and brain functions [1, 2]. Controlling neuronal dynamics via external stimuli provides methods to suppress pathological behavior such as seizures and manipulate brain function [3]. However, it is still challenging to constitute a feedback control law for neurons since we cannot directly observe neuronal multi-dimensional state; we can only obtain a part of the entire neuronal activities. For instance, only noisy membrane potentials are observable [4–6]. Therefore, it is essential to establish a method for simultaneously estimating and controlling the nonlinear neuronal dynamics from only noisy membrane potentials.

To control the neurons, it is required to have accurate information about the state and the dynamics of the neurons. Some previous methods have been proposed to estimate the state and the dynamics of the neurons from lower dimensional observation data based on the statistic machine learning [7–10]. Some previous methods estimate the neuronal dynamics by using offline manner [7–9]. Other previous method estimates the neuronal dynamics by using online manner [10]. On the other hand, some previous method has been proposed to control neurons based on the modern control theory [11, 12] or

stochastic optimal control theory [13]. However, these conventional methods for controlling neuronal dynamics assume either the state or the dynamics to be known. This is unrealistic assumption in neuronal systems, since several neuron models, like the Morris-Lecar model [14], have unobservable latent variables. Moreover, imaging technique provides only lower dimensional variables compared with high dimensional latent variables. Nonetheless, there is no integrated framework for simultaneously estimating and controlling neuronal dynamics.

In this study, we propose a sequential Monte Carlo based framework for simultaneously estimating and controlling the nonlinear individual neuronal state and dynamics under a realistic situation where only noisy membrane potentials can be obtained. We focus on a probabilistic framework to estimate and control neuronal dynamics, in particular, the state-space model and the sequential Monte Carlo (SMC) method as well as model predictive control theory. This paper is organized as follows. We derive the automatic framework to simultaneously estimate and control the single neuron state based on the sequential Monte Carlo method [15, 16], the expectation-maximization algorithm [17], and the SMC-based model predictive control [18] in section 2. In the proposed method, we formulate the state-space model based on the Morris-Lecar neuron model [14] in a partial observation situation. Next, we derive an online algorithm for simultaneously estimating latent variables and parameters by employing the sequential Monte Carlo method and the expectation-maximization algorithm. Then, by combining the model predictive control with the sequential Monte Carlo method, we constitute the automatic feedback control law for the single neuron. In order to verify the effectiveness of the proposed method, we estimate and control the neuron model under simulation environments in section 3. We show the proposed method can successfully estimate and control the neuron state. The concluding remarks are given in section 4.

2. Proposed method

We propose a statistical machine learning based framework for estimating and simultaneously controlling the nonlinear dynamics of individual neurons. The overview of the framework is shown in Fig. 1. Our method consists of three parts; (A) estimating the filtering distribution of the hidden state from partial observation data, (B) estimating the parameter governing the neuronal dynamics, (C) formulating the optimal control problem as a Bayesian statistic problem.

2.1 Morris-Lecar neuron model

Here we formulate a nonlinear state-space model based on the Morris-Lecar neuron model. The Morris-Lecar neuron model is defined as follows:

$$
\begin{aligned}
C_m \frac{dv}{dt'} &= I - g_L(v - E_L) - g_{Ca} m_\infty(v)(v - E_{Ca}) - g_K n(v - E_K), \\
\frac{dn}{dt'} &= -\phi \cosh\left(\frac{v - V_3}{2V_4}\right)(n - n_\infty(v)),
\end{aligned}
\tag{1}
$$

where t' means continuous time, v is a membrane voltage, n is a normalized channel variable, and I is an applied current. Additionally, $m_\infty(v)$ and $n_\infty(v)$ are nonlinear functions given as follows:

$$
\begin{aligned}
m_\infty(v) &= \frac{1}{2}\left\{1 + \tanh\left(\frac{v - V_1}{V_2}\right)\right\}, \\
n_\infty(v) &= \frac{1}{2}\left\{1 + \tanh\left(\frac{v - V_3}{V_4}\right)\right\}.
\end{aligned}
\tag{2}
$$

Other model parameters are described in Table I. The Morris-Lecar model can be expressed by the state vector $x = \begin{bmatrix} v & n \end{bmatrix}^\top$ and the parameter $\theta = \begin{bmatrix} g_L & g_{Ca} & g_K \end{bmatrix}^\top$ as follows:

$$
\frac{dx}{dt'} = f(x, I; \theta) = \begin{bmatrix} \frac{1}{C_m}(I - V_{\mathrm{ion}}(x)^\top \theta) \\ -\phi \cosh\left(\frac{v - V_3}{2V_4}\right)(n - n_\infty(v)) \end{bmatrix},
\tag{3}
$$

where $V_{\mathrm{ion}}(x) = \begin{bmatrix} v - E_L & m_\infty(v)(v - E_{Ca}) & n(v - E_K) \end{bmatrix}^\top$.
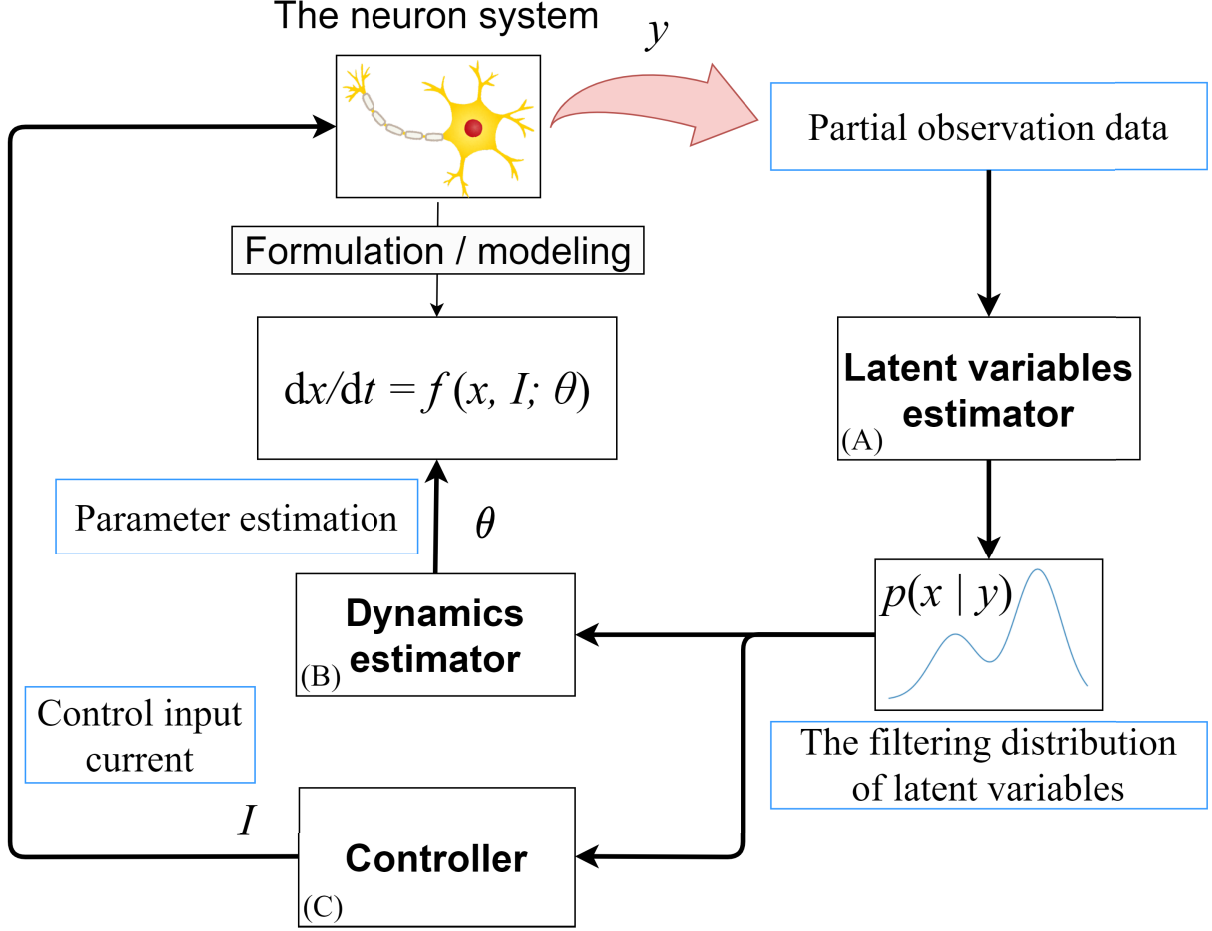
Fig. 1. The overview of the proposal method. (A) We apply the sequential Monte Carlo method in order to estimate the latent state x from the partial, noisy data y. (B) We adapt the expectation-maximization algorithm into online one for estimating the dynamics parameter θ. (C) We apply the sequential Monte Carlo based model predictive control for controlling the neuronal state.

Table I. The parameters for Morris-Lecar model.

parameters	meaning
g_L	conductance of leak current
g_{Ca}	conductance of calcium ion channel
g_K	conductance of potassium ion channel
E_L, E_{Ca}, E_K	reversal potentials
C_m	membrane capacitance
ϕ	opening velocity for the K channel
V_1, V_2, V_3, V_4	constants s.t. $m_\infty(V_1) = n_\infty(V_3) = \frac{1}{2}$

Here, we derive the system model and the observation model for the state-space model. At first, we discretize the continuous-time model given by Eq. (3) with the time step Δt.

$$x_{t+1} = x_t + f(x_t, I_t; \theta)\Delta t, \tag{4}$$

where t denotes a discrete time index, x_t the state value at time $t' = t\Delta t$, respectively. Then, we consider a fluctuating system noise governed by an additive Gaussian noise with zero mean and its covariance matrix $\Sigma_x = \text{diag}(\sigma_v^2, \sigma_n^2) \in \mathbb{R}^{2\times2}$. The system model of the state-space model is formulated as follows:

$$p(x_{t+1} \mid x_t, I_t, \theta) = \mathcal{N}(x_{t+1} \mid x_t + f(x_t, I_t; \theta)\Delta t, \Sigma_x), \tag{5}$$

where $\mathcal{N}(x \mid \mu, \Sigma)$ denotes a probability density function for Gaussian distribution with its mean μ

and its covariance Σ.

Next, we consider the observation model for the state-space model. We assume we can obtain only a one-dimensional time series of noisy membrane potentials $\{y_t\}_{t \geq 0}$. By introducing the additive Gaussian noise with its variance $\sigma_y^2 \in \mathbb{R}$, we formulate the observation model of the state-space model as follows:

$$p(y_t \mid x_t) = \mathcal{N}(y_t \mid g(v_t), \sigma_y^2), \tag{6}$$

where $g(v)$ is arbitrary function which can be either nonlinear or linear. We simply assume g is identity function i.e. $g(v) = v$.

2.2 Sequential Monte Carlo method for estimating latent state

To estimate the latent state x_t from noisy time-series membrane potentials $Y_t = \{y_1, y_2, \ldots, y_t\}$ online, we apply the sequential Monte Carlo (SMC) method [15]. By using SMC we estimate the latent variable from partial observation data, and even if the model includes nonlinearity, the latent variable can be successfully estimated [19].

Here, we consider estimating the latent state x_t by using predictive distribution $p(x_t \mid Y_{t-1})$ and the filtering distribution $p(x_t \mid Y_t)$. In the SMC method, both distributions are approximated by using particles, which represent the value of the latent state. The SMC method is divided into two steps: the prediction step and filtering step.

In the predictive step, we sample the particles with the system model [Eq. (5)] as follows:

$$\bar{x}_t^{(i)} \sim p(x_t \mid x_{t-1}^{(i)}, I_{t-1}, \theta) \quad \text{for } i = 1, \ldots, N, \tag{7}$$

where i is the index of the particle and N is the number of the particles. $\bar{x}_t^{(i)}$ represents the predictive particles which approximate the predictive distribution of the latent variable. In the filtering step, we resample the particles with the observation model [Eq. (6)]. First, the likelihood, called weight, of each ith particle is calculated and normalized as follows:

$$\hat{w}_t^{(i)} = p(y_t \mid \bar{x}_t^{(i)}), \quad \bar{w}_t^{(i)} = \frac{\hat{w}_t^{(i)}}{\sum_{i=1}^N \hat{w}_t^{(i)}}. \tag{8}$$

Then, each particle is resampled based on the likelihoods. We use the multinomial resampling [20, 21] given as following procedure for each particle:

$$a^{(i)} \quad \sim \quad \mathcal{M}\left(\bar{w}_t^{(1)}, \ldots, \bar{w}_t^{(N)}\right), \tag{9}$$

$$x_t^{(i)} \quad = \quad \bar{x}_t^{(a^{(i)})}, \tag{10}$$

$$w_t^{(i)} \quad = \quad \frac{1}{N}, \tag{11}$$

where $\mathcal{M}\left(\bar{w}_t^{(1)}, \ldots, \bar{w}_t^{(N)}\right)$ denotes a multinomial distribution supported on $\{1, 2, \ldots, N\}$. By using resampled particles and weights, we obtain an approximated filtering distribution as follows:

$$p(x_t \mid Y_t, \theta) \approx \sum_{i=1}^N w_t^{(i)} \delta(x_t - x_t^{(i)}), \tag{12}$$

where $\delta(\cdot)$ is a Dirac's delta function. The obtained particles $\{(x_t^{(i)}, w_t^{(i)})\}_{i=1}^N$ also are used not only to estimate the parameters in section 2.3, but also to constitute the feedback law introduced in section 2.4.

2.3 Online EM algorithm for estimating model parameter

To estimate the model parameter θ online, we adapt the original offline-based expectation-maximization (EM) algorithm into an online one with the SMC. The original EM algorithm is an offline algorithm that updates the parameter θ iteratively with the following rule:

$$\theta_k = \arg\max_\theta E\left[\log p(Y_{1:T}, X_{1:T} \mid \theta)\right]_{p(X_{1:T} \mid Y_{1:T}, \theta_{k-1})}, \tag{13}$$

where k is an iteration number, $X_{i:j}$ and $Y_{i:j}$ respectively denote the time series of the latent variable and observations $X_{i:j} = \{x_i, \ldots, x_j\}$, $Y_{i:j} = \{y_i, \ldots, y_j\}$, and T is a terminal time index.

In our study, we consider the parameter θ_k is updated at times $\{k \times T_{\text{interval}}\}_{k=1,2,\cdots}$, where k represents the renewal number and T_{interval} denotes the interval of the update time. In addition, the expectation with respect to the previous parameter θ_{k-1} in Eq. (13) is replaced with the filtering distribution obtained online via SMC. The procedure is summarized as follows:

$$\hat{\theta}_k = \arg\max_\theta E\left[\log p(Y_{\text{win}(t)}, X_{\text{win}(t)} \mid \theta)\right]_{p(X_{\text{win}(t)} \mid Y_{\text{win}(t)}, \theta_{k-1})}, \tag{14}$$

$$\theta_k = (1 - \eta_k)\theta_{k-1} + \eta_k \hat{\theta}_k, \tag{15}$$

where $X_{\text{win}(t)} = X_{t-L(t):t}$ and η_k is a number that satisfy $0 < \eta_k < 1, \eta_k \to 0$ as $k \to \infty$. $L(t)$ is a function to determine the length of the data used for updating and defined as follows:

$$L(t) = T_{\text{interval}} \left\{ \lfloor k Z_1 \exp(-\tau_{\text{win}} k) \tanh(k) + Z_2 \tanh(\tau_{\text{win}} k) \rfloor + 1 \right\}, \tag{16}$$

where $k = \lfloor T_{\text{interval}}/t \rfloor$, Z_1, Z_2, and τ_{win} are constants, and $\lfloor \cdot \rfloor$ is the floor function.

Here, we derive an update equation for the Morris-Lecar neuron-based state-space model given by Eqs. (5) and (6). After calculation, we obtain the following explicit update rule:

$$\hat{\theta}_k = E[-A^{-1}(X_{\text{win}(t)})b(X_{\text{win}(t)})]_{p(X_{\text{win}(t)} \mid Y_{\text{win}(t)}, \theta_{k-1})}, \tag{17}$$

where $A(X_{\text{win}(t)}), b(X_{\text{win}(t)})$ are defined as follows:

$$A(X_{\text{win}(t)}) = \left(\frac{\Delta t}{C_m}\right)^2 \sum_{\tau \in \mathcal{T}(t)} V_{\text{ion}}(x_\tau) V_{\text{ion}}(x_\tau)^\top, \tag{18}$$

$$b(X_{\text{win}(t)}) = \frac{\Delta t}{C_m} \sum_{\tau \in \mathcal{T}(t)} \left(v_{\tau+1} - v_\tau - \frac{\Delta t}{C_m} I_\tau\right) V_{\text{ion}}(x_\tau). \tag{19}$$

Here, $\mathcal{T}(t) = \{t - L(t), \ldots, t - 1\}$. We consider applying this update procedure for a fixed times n_{iter} in the experiments.

2.4 Model predictive control driven by the sequential Monte Carlo method

To constitute the feedback control law under a practical situation where the state and the parameters are unknown, we combine the SMC method-based model predictive control (SMC-MPC) [18] with our dynamics estimator. The model predictive control (MPC) in a modern control theory is a feedback control law that (1) at time t, formulates an optimal control problem in a future interval called horizon with the given model $x_{t+1} = f(x_t, I_t)$, (2) solves the optimal problem to obtain the optimal control input series $\{I_\tau^*\}_{\tau \in \text{horizon}}$ in the horizon, (3) adopts the first one of the obtained optimal control input series. Here, the optimal problem is attributed to solving the Euler-Lagrange equation which includes the derivative of the model with respect to the state and control input. On the other hand, in the SMC-MPC, the optimal control problem is replaced with a filtering problem based on an extended state-space model.

Figure 2 shows the overview of the SMC-MPC. In our framework, two distinct SMC modules exist; one (SMC) is a state estimator described in the subsection 2.2 and the other (second SMC) is a solver of the optimal control problem defined below. Once the state particles $\left\{x_t^{(i)}, w_t^{(i)}\right\}_{i=1}^N$ are obtained via SMC, we sample the control input particles $\left\{\bar{I}_t^{(i)}\right\}_{i=1}^N$ by using the input proposal distribution $p\left(\bar{I}_t \mid I_{t-1}\right)$. With the extended particles $\left\{x_t^{(i)}, \bar{I}_t^{(i)}, w_t^{(i)}\right\}_{i=1}^N$ and the reference trajectory $\{r_t, r_{t+1}, \ldots, r_{t+T_H}\}$, we solve the control-filtering problem by applying the second SMC in the horizon. Finally, we adopt the actual control input I_t obtained from the filtering distribution of the initial control input $p(\bar{I}_t \mid r_{t:t+T_H})$.
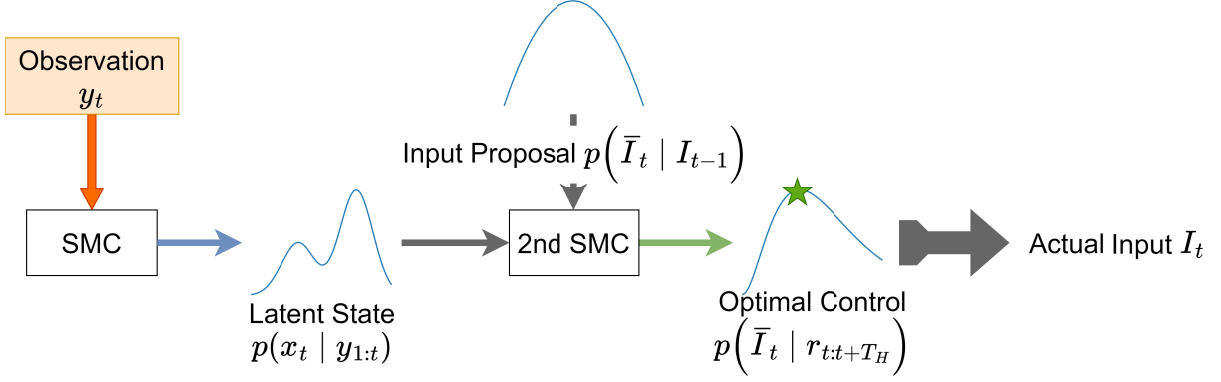
Fig. 2. The overview of the SMC-MPC. The particles and their weights obtained via the state estimator (SMC) is directly used as initial particles by the SMC in the horizon (Second SMC). The input particles are sampled from the proposal distribution. We can obtain the filtering distribution given the reference trajectory and the optimal control input is calculated by some point estimator like mean.

Here, we discuss the detail of the SMC-MPC. To formulate the control-filtering problem, we define the extended state-space model in the horizon. The extended state-space model consists of the extended system model and the extended observation model as usual. To define the extended system model, we consider the control input I_t is regarded as the part of the extended state $\zeta_t = (\bar{x}_t, \bar{I}_t, \tilde{I}_t)$, where $\tilde{I}_t$ is a dummy input to preserve the initial control input. Note that the notation $\bar{x}_t, \bar{I}_t$ is required in order to distinguish the prediction transition $\{(\bar{x}_t, \bar{I}_t), (\bar{x}_{t+1}, \bar{I}_{t+1}), \ldots, (\bar{x}_{t+T_H}, \bar{I}_{t+T_H})\}$ in the horizon and the actual state time-series $\{x_1, \ldots, x_t\}$ and input time-series $\{I_1, \ldots, I_t\}$. Hence we formulate the extended system model in the horizon as follows:

$$\begin{cases} \bar{x}_{\tau+1} \sim p(\bar{x}_{\tau+1} \mid \bar{x}_\tau, \bar{I}_\tau; \theta), \quad \bar{x}_t = x_t \\ \bar{I}_{\tau+1} \sim p(\bar{I}_{\tau+1} \mid \zeta_\tau), \quad \bar{I}_t \sim p(I_t \mid I_{t-1}), \\ \tilde{I}_{\tau+1} = \tilde{I}_\tau, \quad \tilde{I}_t = \bar{I}_t, \end{cases} \quad (20)$$

where $\tau \in \{t, t+1, \ldots, t+T_H\}$ is the discrete time in the horizon and T_H is a length of the horizon. Here, the state transition $p(\bar{x}_{\tau+1} \mid \bar{x}_\tau, \bar{I}_\tau; \theta)$ in the horizon is defined by using the original system model $p(x_{t+1} \mid x_t, I_t; \theta)$ given by Eq. (5). In our study, the control input transition $p(\bar{I}_{\tau+1} \mid \zeta_\tau)$ and the initial distribution for the control input $p(\bar{I}_t \mid I_{t-1})$ are designed to ensure that the input time-series $\{\bar{I}_t, \ldots, \bar{I}_{t+T_H}\}$ always remain within a certain range as follows:

$$\bar{I}_{\tau+1} = \text{clip}_{I_{\text{lim}}}(\bar{I}_\tau + \bar{I}_{\tau+1}^{\text{stream}} + z_\tau) - \bar{I}_{\tau+1}^{\text{stream}}, \quad (21)$$

$$\bar{I}_t = \text{clip}_{I_{\text{lim}}}(I_{t-1} + \bar{I}_t^{\text{stream}} + \alpha z_{t-1}) - \bar{I}_t^{\text{stream}}, \quad (22)$$

where $z_{t-1} \sim \mathcal{N}(0, \sigma_I^2)$, α is a scaling constant, $\bar{I}_t^{\text{stream}}$ is a sum of external currents streaming from other neurons, $0 < I_{\text{lim}}$ is a limit of the total applied currents, and clip is a clamp function defined as $\text{clip}_a(u) = \{|u + a| - |u - a|\}/2$. In our study, the prediction of the external currents $\bar{I}_\tau^{\text{stream}}$ is fixed at the value observed at time t, i.e. $\bar{I}_\tau^{\text{stream}} = I_t^{\text{stream}}$ for $\tau \in \{t, t+1, \ldots, t+T_H\}$. Note that the transition probability is not explicitly required to be known in the extended state-space model.

Next, we formulate the extended observation model in the horizon. We consider controlling the state towards the given reference membrane potentials $\{r_\tau\}_{\tau=t}^{t+T_H}$. In the SMC-MPC, we regard the reference trajectory as a realization of the Markov chain with the extended system model in the horizon. Therefore, we formulate the extended observation model (referred as a setpoint equation in [18]) as follows:

$$r_\tau \sim p(r_\tau \mid \zeta_\tau) = \mathcal{N}(r_\tau \mid \bar{v}_\tau, \sigma_r^2), \quad (23)$$

where σ_r^2 is a variance which adjusts an acceptable error against the reference trajectory.

By using SMC in the horizon which we call the *second SMC*, we can efficiently solve the filtering problem defined by Eqs. (20) and (23) with given observation data $\{r_\tau\}_{\tau=t}^{t+T_H}$. Here, to initialize

the extended particles $\{\zeta_t^{(i)}, w_t^{(i)}\}_{i=1}^N$, we use the state particles $\{(x_t^{(i)}, w_t^{(i)})\}_{i=1}^N$ obtained via the state estimation. After applying Second SMC in the horizon, we obtain the filtered extended particles $\{\zeta_{t+T_H}^{(i)}, w_{t+T_H}^{(i)}\}_{i=1}^N$. By extracting the dummy input particles inheriting initial inputs and their weights $\{\tilde{I}_{t+T_H}^{(i)}, w_{t+T_H}^{(i)}\}_{i=1}^N$, we obtain the point estimation of this filtering distribution, e.g.

$$I_t^* = \sum_{i=1}^N w_{t+T_H}^{(i)} \tilde{I}_{t+T_H}^{(i)}. \tag{24}$$

Finally, we adopt $I_t = I_t^*$ as the actual control input at time t.

Here we briefly touch the difference between the conventional MPC and the SMC-MPC. In the SMC-MPC, it is not necessary to derive derivatives of the model. Moreover, we do not estimate the state explicitly but need only the particles. The only requirements of the SMC-MPC itself is that the model can be simulated in forward and this requirements is satisfied in typical cases. On the other hand, the conventional MPC requires derivatives of the Hamiltonian and point-estimation of the state, which leads inflexibility. Nevertheless, the control input obtained by applying the SMC-MPC is equivalent to the value obtained by using the conventional MPC under some assumptions [18].

We have proposed an automatic control framework, even if the state and the dynamics are unknown. Although the conventional MPC enables us to handle even a nonlinear model, the conventional MPC requires the state and the model to be known. The SMC-MPC [18] resolves one side of this problem by combining the SMC method with MPC effectively. However, they assumed that the model dynamics to be known. The proposed method constitutes the feedback control law by combining the SMC-MPC with the state and the parameter estimators.

3. Experiment

3.1 Setting

We verify the effectiveness of our proposal method using simulation environments. The true neuronal dynamics is assumed to be the Morris-Lecar neuron model [22]. Here, we assume multi-dimensional latent variables v and n, and the conductances of ion channels g_L, g_{Ca}, g_K are unknown. The reference membrane potentials are generated by using the Morris-Lecar model in advance. In practice, the constant current influenced by other neurons exists. We consider that the net input current I for the neuron consists of the constant current I_{stream} and the controllable input I_{in}, i.e. $I = I_{stream} + I_{in}$. We assume the sum of external currents streaming from other neurons I_t^{stream} to be known.

In the following subsections, we set $\Delta t = 0.1$, the covariance of the system noise $\Sigma_x = \text{diag}(\sigma_v^2, \sigma_n^2) = \text{diag}(0.1^2, 0.001^2)$ and the variance of the observation noise $\sigma_y^2 = 0.1^2$. The number of particles N is set to be $N = 2000$ in subsection 3.2 and $N = 1000$ in subsection 3.3, respectively. For the state estimation, we simply use weighted mean of the particles as a point estimation at time t i.e. $x_{est,t} = \sum_{i=1}^N w_t^{(i)} x_t^{(i)}$. For the online parameter estimation, we set $T_{interval} = 100, n_{iter} = 20, \eta_k = \frac{1}{\sqrt{k+1}}$, $Z_1 = \frac{2}{3}, Z_2 = 3, \tau_{win} = 0.1$. For SMC-MPC, the length of the horizon T_H and the acceptable error variance σ_r^2 is set to be $T_H = 10, \sigma_r^2 = 0.1^2$, respectively. We set the variance $\sigma_I^2 = 1$ and the initial scaling constant $\alpha = 10$ in the input proposal. As the constraint with respect to the net input for the neuron, we set the maximum norm of the net input $I_{lim} = 150$.

3.2 Result for estimating and controlling the neuronal dynamics: case where the reference is the steady firing

Figure 3 shows the simulation result where the true parameter is assumed to be SNLC parameter [22]. Here, we verify whether our proposal framework can estimate the neural hidden state and its parameters from only noisy membrane potentials, but also simultaneously control the membrane potentials towards the desired ones. In this subsection, we consider the reference time-series v_{ref} is a steady firing. The middle left graph of Fig. 3 shows the true membrane potential v_{true} (blue line), the estimated one v_{est} (orange dashed line), and the reference one v_{ref} (green dotted line). We can see the estimated membrane potential tracks the true value as time passes. Here, the gray dash-dotted line
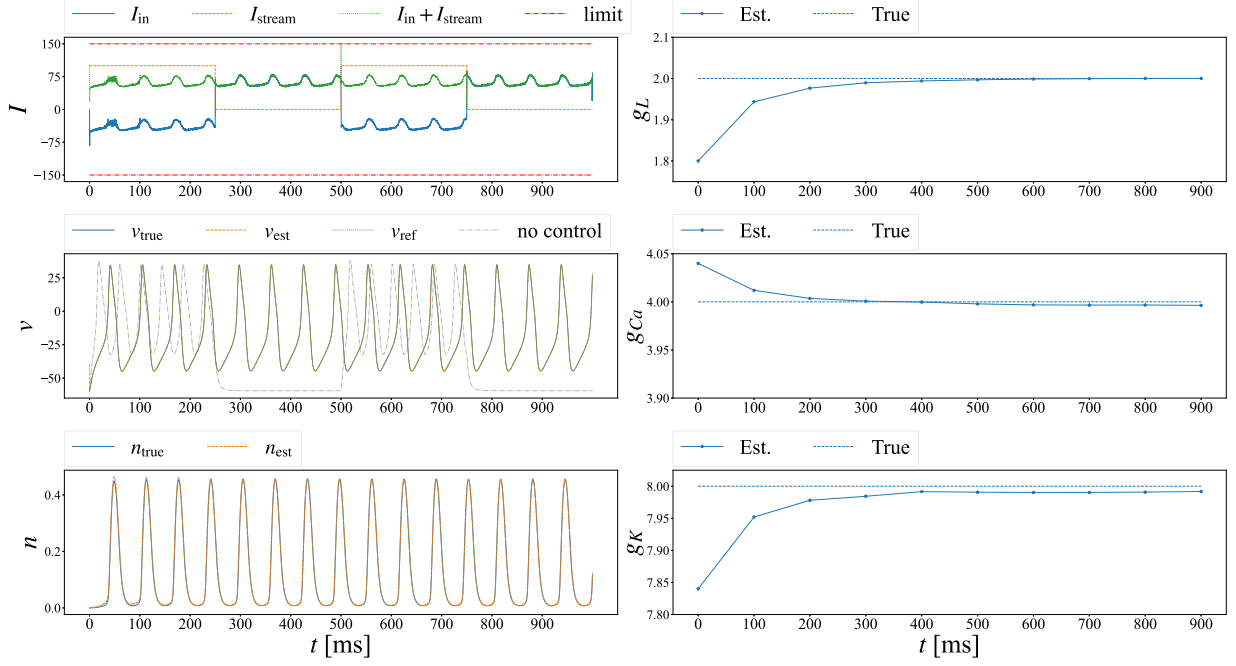
Fig. 3. Result for estimating neuronal dynamics and controlling the membrane potentials towards the steady firing state by using our proposed method. Left: currents including control current, membrane potential v, and channel variable n. Right: membrane conductances g_L, g_{Ca}, and g_K. By applying our proposed framework, the neuronal activity is modulated towards the periodic firing pattern.

represents the simulation result if no input current is applied. Without our proposal method, the neuron does not exhibit the steady firing but periodically switches the firing and resting state. On the other hand, by applying our proposed method, the membrane potentials tracks the time-varying reference membrane potentials and as a result, the neuron gets to represent the steady firing pattern. The lower left graph of Fig. 3 shows the true channel variable n_{true} (blue line) and the estimated value n_{est} (orange dashed line). Note that n is unobservable. We also find that the estimated value of the channel variable tracks the true value.

Next, we clarify whether our method can estimate and control the neuronal dynamics using feedback input current with a limited range of strength. The upper left graph of Fig. 3 shows the applied input current I_{in} (blue line), the streaming current I_{stream} (orange dotted line), and the limit I_{lim} (red dash-dotted line). Here, I_{in} is automatically calculated by the proposed framework. We can see the sum of the control and streaming current (green dashed line) satisfies $|I_{\mathrm{in}} + I_{\mathrm{stream}}| \leq I_{\mathrm{lim}}$. Hence, we can simultaneously control the neuronal dynamics even if we impose the input constraint.

Finally, we clarify whether our method can estimate the unknown parameters. The graphs on the right side in Fig. 3 show the estimated parameters and their true values. We can see each estimated conductance value converges to the true value as the online-adapted EM algorithm is applied.

3.3 Result for estimating and controlling the neuronal dynamics: case where the reference includes transition between complex firing and resting state

Figure 4 shows another experiment result where the true parameter is set to be homoclinic parameter [22]. We confirm here that our method can estimate and control the neuronal dynamics even if the parameter set is different. In this subsection, we consider the reference time-series v_{ref} includes the transition between complex firing state and resting state. That is, the reference v_{ref} is assumed to be a firing state for $0 \leq t \leq 250$ and $500 \leq t \leq 750$ while resting state for $250 \leq t \leq 500$ and $750 \leq t \leq 1000$, and there are switches between two different states at $t = 250, 500$ and 750. The middle left graph of Fig. 4 shows the true membrane potential v_{true} (blue line), the estimated one v_{est} (orange dashed line), and the reference one v_{ref} (green dotted line). We can see the estimated
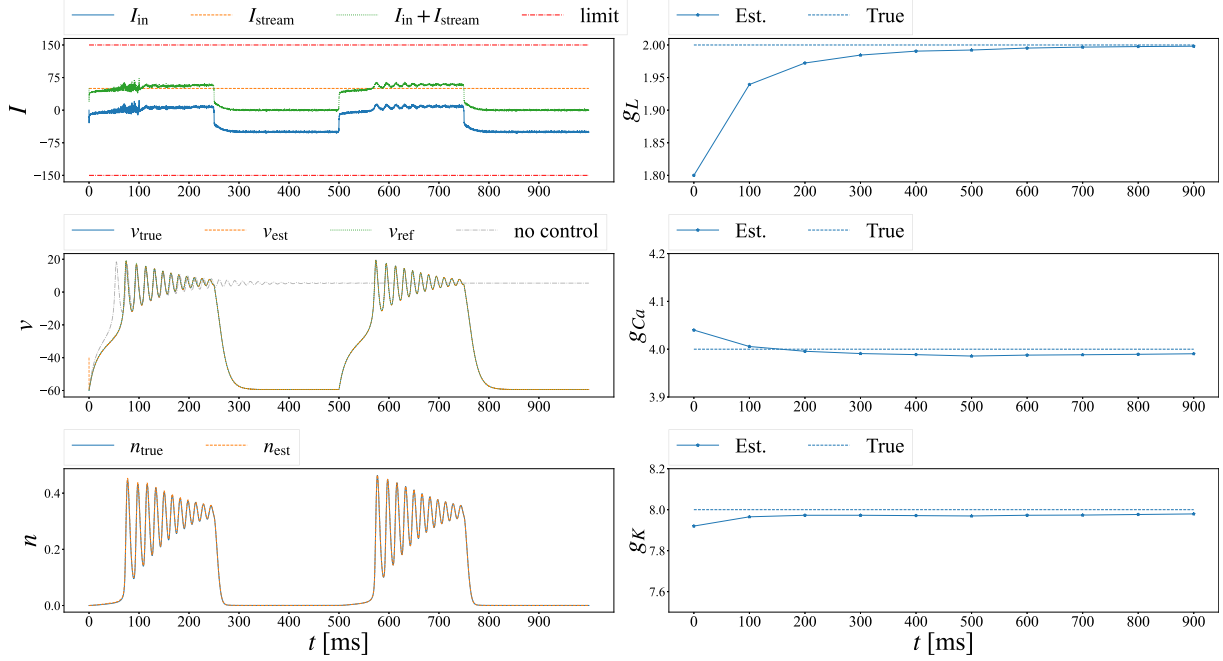
Fig. 4. Result for estimating neuronal dynamics and controlling the membrane potentials from the resting state to the firing state and vice versa by applying the proposed method. Left: currents including control current, membrane potential v, and channel variable n. Right: membrane conductances g_L, g_{Ca}, and g_K. By applying our proposed framework, the neuronal activity is controlled from resting to complex firing patterns and vice versa.

membrane potential tracks the true value as time passes, even when the control reference includes the state transition between the firing and resting state. Moreover, we can control the neuronal membrane potentials towards the desired firing pattern by using our proposal method. Note that the control input (the upper left graph) always satisfies the norm constraint. The lower left graph of Fig. 4 shows the true channel variable n_{true} (blue line) and the estimated value n_{est} (orange dashed line). We also find that we can successfully estimate the latent variables, even if the neuron represents the different firing pattern.

Next, we clarify whether our method can estimate and control the neuronal dynamics using feedback input current with the norm constraint. The upper left graph of Fig. 4 shows the applied input current I_{in} (blue line), the streaming current I_{stream} (orange dotted line), and the limit I_{lim} (red dash-dotted line). We can see the sum of the control and streaming current (green dashed line) satisfies $|I_{\text{in}} + I_{\text{stream}}| \leq I_{\text{lim}}$. Hence, we can simultaneously control the neuronal dynamics even if we consider the different parameter set.

Finally, we clarify whether our method can estimate the unknown parameters. The graphs on the right side in Fig. 4 show the estimated parameters and their true values. We can see that the estimated parameters converge to the true value as time passes.

The result indicates that our proposed framework can simultaneously estimate the latent state and the parameters and control the membrane potential towards the desired ones, even if the conductances are unknown. In particular, to clarify the effectiveness of the proposed framework, we have shown that our proposed framework can modulate neuronal activities towards periodic firing in Fig. 3 and reference trajectory including transition between complex firing and resting state in Fig. 4.

3.4 Robustness against the observation noise

Here, we discuss the robustness of our proposed framework. In the previous subsections, we verified our proposal framework under the observation noise $\sigma_y = 0.1$. Although we can obtain relatively precise membrane potentials by using recent imaging technology, it is worth considering the situation where the observation noise is large. We evaluate our proposed framework under different noise levels.
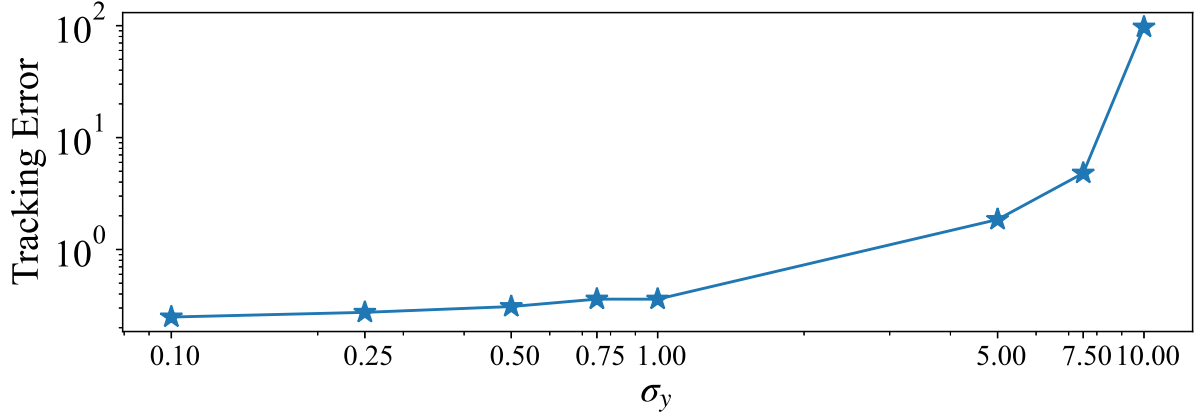
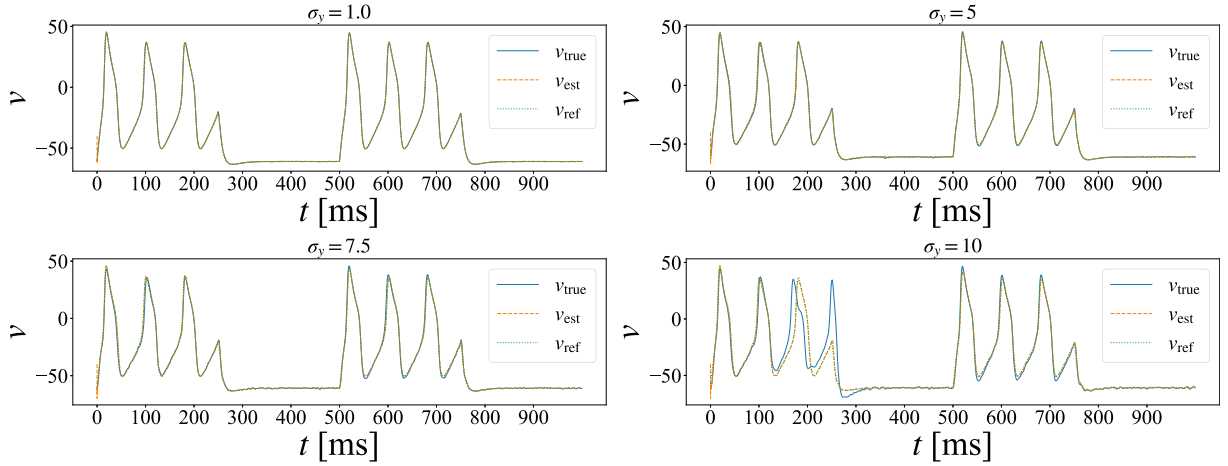**Fig. 5.** Comparison of the tracking error for different levels of observation noises.

Fig. 6. Comparison of the behavior of membrane potentials for different levels of observation noises. Our proposed framework can successfully estimate and control the neuronal dynamics even in noisy environment ($\sigma_y \lesssim 7.5$). In addition, our proposed framework can almost estimate and control the neuronal dynamics even in extremely noisy environment ($\sigma_y = 10$).

Here, the true parameter is set to be the parameter for Hopf bifurcation [22].

Figure 5 shows how the tracking error changes as the observation noise σ_y becomes larger. Here, the tracking error E_{tracking} is calculated as follows:

$$E_{\text{tracking}} = \frac{1}{T} \sum_{t=1}^{T} \{v_{\text{true}}(t) - v_{\text{ref}}(t)\}^2 \tag{25}$$

In Fig. 5, we consider several noise levels, $\sigma_y = 0.1, 0.25, 0.5, 0.75, 1, 5, 7.5,$ and 10. Note both axis in Fig. 5 are log-scale. The obtained tracking errors for $\sigma_y \leq 5$ are very small, i.e., $E_{\text{tracking}} \leq 2.0$. On the other hand, the error for $\sigma_y = 10$ almost reaches 10^2.

To confirm to what extent the observation noise affects our automatic control strategy, we visualize the membrane potentials in the experiment. Figure 6 shows the behavior of the membrane potentials for specific observation noises. In Fig. 6, the upper graphs correspond to moderate noise levels ($\sigma_y = 1, 5$) and the lower graphs strong noise levels ($\sigma_y = 7.5, 10$). Concretely, the left up graph in Fig. 6 corresponds to the experiment result when $\sigma_y = 1.0$, the right up $\sigma_y = 5$, the left down graph $\sigma_y = 7.5$, and the right down one $\sigma_y = 10$, respectively. Each graph shows the true membrane potential v_{true} (blue line), the estimated one v_{est} (orange dashed line), and the reference one v_{ref} (green dotted line) that are obtained for the corresponding noise level. We can see our proposed framework controls the spiking time precisely given by the reference membrane potentials under noisy environment, i.e. $\sigma_y \leq 7.5$. On the other hand, in the right down graph ($\sigma_y = 10$) in Fig. 6, our

proposed framework make finite errors for $150 \leq t \leq 300$, whereas the true membrane potentials are traced precisely in the other time period ($0 \leq t \leq 150, 300 \leq t \leq 900$) and the dynamical membrane behavior of polarization and hyperpolarization can be traced even for $150 \leq t \leq 300$. This error for $150 \leq t \leq 300$ is because extremely high noise disturbs the state and dynamics estimators, which causes the controller to estimate an non-optimal control input.

With recent advances in imaging technology, we can observe neuronal membrane potentials more precisely [23]. In other words, the observation noise will become less and less. The experiment shows that our proposed framework can successfully estimate and control the neuronal dynamics when $\sigma_y \leq 7.5$. Hence our proposed framework is applicable in realistic situation and sufficiently robust against the observation noise.

4. Conclusion

In this paper, we propose a statistical machine learning based framework for estimating the neuronal state and dynamics, and simultaneously controlling the neuronal dynamics. We have shown that the proposal method not only well estimates the multi-dimensional hidden state and the parameters of the neuron, but also successfully control the membrane potentials towards the desired ones under a partial observation situation.

It should be noted that our framework can handle the nonlinearity of the model without derivatives for control thanks to SMC-MPC. Hence, our framework is applicable to general parametric state-space models, not limited for the Morris-Lecar neuronal model. While our framework is rather general for various types of state-space models described by parametric model formulation, it is essential to extend our framework into the nonparametric models formulation. In the future work, it would be important to consider the case where the internal fluctuation is non-Gaussian or chaotic noise.

Acknowledgments

References

[1] J. Shine, E. Müller, B. Munn, J. Cabral, R. Moran, and M. Breakspear, "Computational models link cellular mechanisms of neuromodulation to large-scale neural dynamics," *Nature Neuroscience*, vol. 24, pp. 765–776, 2021.

[2] D. Chialvo, "Emergent complex neural dynamics," *Nature Physics*, vol. 6, pp. 744–750, 2010.

[3] L. Fenno, O. Yizhar, and K. Deisseroth, "The development and application of optogenetics," *Annual Review of Neuroscience*, vol. 34, pp. 389–412, 2011.

[4] N. Vogt, "Voltage imaging in vivo," *Nature Reviews Neuroscience*, vol. 16, p. 573, 2019.

[5] T. Knöpfel and C. Song, "Optical voltage imaging in neurons: moving from technology development to practical tool," *Nature Reviews Neuroscience*, vol. 20, pp. 719–727, 2019.

[6] D. Peterka, H. Takahashi, and R. Yuste, "Imaging voltage in neurons," *Neuron*, vol. 69, pp. 9–21, 2011.

[7] T. Omori, "Estimating nonlinear spatiotemporal membrane dynamics in active dendrites," *Neural Information Processing*, vol. 8834, pp. 27–34, 2014.

[8] T. Omori and K. Hukushima, "Extracting nonlinear spatiotemporal dynamics in active dendrites using data-driven statistical approach," *Journal of Physics: Conference Series*, vol. 699, p. 012011, 2016.

[9] H. Inoue, K. Hukushima, and T. Omori, "Estimation of neuronal dynamics of Izhikevich neuron models from spike-train data with particle Markov chain Monte Carlo method," *Journal of the Physical Society of Japan*, vol. 90, no. 10, pp. 1–12, 2021.

[10] S. Ditlevsen and A. Samson, "Estimation in the partially observed stochastic Morris-Lecar neuronal model with particle filter and stochastic approximation methods," *The Annals of Applied Statistics*, vol. 8, pp. 674–702, 2014.

[11] S. Vaidyanathan, "Adaptive control of the FitzHugh-Nagumo chaotic neuron model," *International Journal of PharmTech Research*, vol. 8, no. 6, pp. 117–127, 2015.

[12] G. Ullah and S. Schiff, "Tracking and control of neuronal Hodgkin-Huxley dynamics," *Physical Review. E*, vol. 79, p. 040901, 2009.

[13] A. Iolov, S. Ditlevsen, and A. Longtin, "Stochastic optimal control of single neuron spike trains," *Journal of Neural Engineering*, vol. 11, p. 046004, 2014.

[14] C. Morris and H. Lecar, "Voltage oscillations in the barnacle giant muscle fiber," *Biophysical Journal*, vol. 35, no. 1, pp. 193–213, 1981.

[15] G. Kitagawa, "A Monte Carlo filtering and smoothing method for non-Gaussian nonlinear state space models," *Proc. 2nd U.S.-Japan Joint Seminar on Statistical Time Series*, 1993.

[16] A. Doucet, N. Freitas, and N. Gordon, *Sequenatial Monte Carlo Methods In Practice*, Springer, 2001.

[17] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[18] D. Stahl and J. Hauth, "PF-MPC: particle filter-model predictive control," *Systems and Control Letters*, vol. 60, pp. 632–643, 2011.

[19] T. Omori, T. Kuwatani, A. Okamoto, and K. Hukushima, "Bayesian inversion analysis of nonlinear dynamics in surface heterogeneous reactions," *Physical Review E*, vol. 94, p. 033305, 2016.

[20] R. Douc and O. Cappe, "Comparison of resampling schemes for particle filtering," *ISPA 2005, Proc. 4th International Symposium on Image and Signal Processing and Analysis*, pp. 64–69, 2005.

[21] J. Hol, T. Schon, and F. Gustafsson, "On resampling algorithms for particle filters," *2006 IEEE Nonlinear Statistical Signal Processing Workshop*, pp. 79–82, 2006.

[22] G. Ermentrout and D. Terman, *Mathematical Foundations of Neuroscience*, Springer, 2010.

[23] Y. Gong, C. Huang, J. Li, B. Grewe, Y. Zhang, S. Eismann, and M. Schnitzer, "High-speed recording of neural spikes in awake mice and flies with a fluorescent voltage sensor," *Science*, vol. 350, pp. 1361–1366, 2015.