RESEARCH

CrossMark

Sequential estimation of intrinsic activity and synaptic input in single neurons by particle filtering with optimal importance density

Pau Closas[1*] ([iD]) and Antoni Guillamon[2]

Abstract

This paper deals with the problem of inferring the signals and parameters that cause neural activity to occur. The ultimate challenge being to unveil brain's connectivity, here we focus on a microscopic vision of the problem, where single neurons (potentially connected to a network of peers) are at the core of our study. The sole observation available are noisy, sampled voltage traces obtained from intracellular recordings. We design algorithms and inference methods using the tools provided by stochastic filtering that allow a probabilistic interpretation and treatment of the problem. Using particle filtering, we are able to reconstruct traces of voltages and estimate the time course of auxiliary variables. By extending the algorithm, through PMCMC methodology, we are able to estimate hidden physiological parameters as well, like intrinsic conductances or reversal potentials. Last, but not least, the method is applied to estimate synaptic conductances arriving at a target cell, thus reconstructing the synaptic excitatory/inhibitory input traces. Notably, the performance of these estimations achieve the theoretical lower bounds even in spiking regimes.

Keywords: State-space models, Inference and learning, Particle filtering, Synaptic conductance estimation, Spiking neuron, Conductance-based model, Intracellular recording

1 Introduction

Measurements of membrane potential traces constitute the main observable quantities to derive a biophysical neuron model. In particular, the dynamics of auxiliary variables and the model parameters are inferred from voltage traces, in a costly process that typically entails a variety of channel blocks and clamping techniques (see, for instance, [1]), as well as some uncertainty in the parameter values due to noise in the signal. Recent works in the literature deal with the problem of inferring hidden parameters of the model; see, for instance, [2–4] and, for an exhaustive review, [5]. In the same line, attempts to extract connectivity in networks of neurons from calcium imaging, see [6], are worth mentioning.

Apart from inferring intrinsic parameters of the model, voltage traces are also useful to obtain valuable information about synaptic input, an inverse problem with some satisfactory (see, for instance, [7–13]) but no complete

solutions yet. The main shortcomings are the requirement of multiple (supposedly identical) trials for some methods to be applied and the need of avoiding signals obtained when ionic currents are active. The latter constraint arises from the fact that many methods rely on the linearity of the signal and this is not possible to achieve under quite general situations, like spiking regimes (see [14]) or subthreshold regimes when specific currents (e.g., AHP, LTS) are active (see [15]).

The problem investigated in this paper considers recordings of noisy voltage traces to infer the hidden gating variables of the neuron model, as well as filtered voltage estimates, model parameters, and input synaptic conductances.

Figure 1 shows the basic setup we are dealing with in this article. The neuron under observation has its own dynamics, producing electrical voltage patterns. The generation of action potentials is regulated by internal drivers (e.g., the active gating variables of the neuron) as well as exogenous factors like excitatory and inhibitory synaptic conductances produced by pools of connected neurons. This system is unobservable, in the sense that we cannot

*Correspondence: closas@northeastern.edu
[1]Northeastern University, Boston, MA 02115, USA
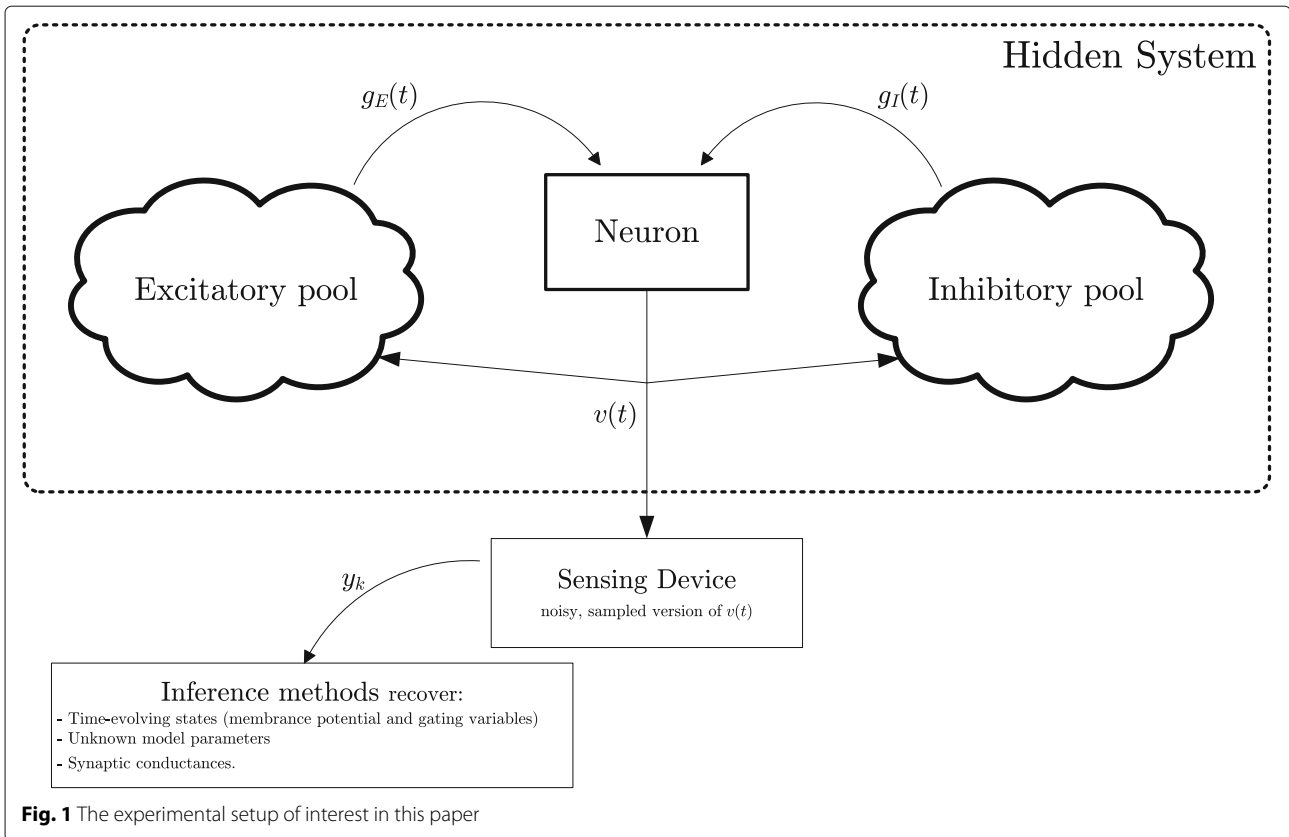Full list of author information is available at the end of the article

Fig. 1 The experimental setup of interest in this paper

measure it directly. The sole observation from this system are the noisy membrane potentials y_k. In this experimental scenario, the ultimate goal is to extract the following quantities:

1. The time-evolving states characterizing the neuron dynamics, including a filtered membrane potential and the dynamics of the gating variables
2. The parameters defining the neuron model
3. The dynamics of synaptic conductances and its parameters, the final goal being to discern the temporal contributions of global excitation from those of global inhibition, $g_E(t)$ and $g_I(t)$, respectively.

An ideal method should be able to sequentially infer the time course of the membrane potential and its intrinsic/extrinsic activity from noisy observations of a voltage trace. The main features of the envisaged algorithm are fivefold: (i) *Single-trial*: the method should be able to estimate the desired signals and parameters from a single voltage trace, thus avoiding the experimental variability among trials; (ii) *Sequential*: the algorithm should provide estimates each time a new observation is recorded, thus avoiding re-processing of all data streams each time; (iii) *Spike regime*: contrary to most solutions operating only under the subthreshold assumption, the method should

be able to operate in the presence of spikes as well; (iv) *Robust*: if the method is model-dependent, thus implying knowledge of the model parameters, then the algorithm should be provided with enhancements to adaptively learn these parameters; and (v) *Statistically efficient*: the performance of the method should be close to the theoretical lower bounds, meaning that the estimation error cannot be substantially reduced. Notice that the focus here is not on reducing the computational cost of the inference method, and thus, we allow ourselves to use resource-consuming algorithms. Indeed, the target application does not demand (at least as a main requirement) real-time operation, and thus, we prioritized performance (i.e., estimation accuracy and the rest of features described earlier) in our developments.

According to the above desired features, in this work, that substantially extends our previous contributions [16, 17], we are interested in methods that can provide estimations from a single trial and avoid the need of repetitions that could be contaminated by neuronal variability. Particularly, we concentrate on methods to extract intrinsic activity of ionic channels, namely the probabilities of opening and closing ionic channels, and the contribution of synaptic conductances. We built a method based on Bayesian theory to sequentially infer these quantities from single-trace, noisy membrane potentials. The material

therein includes a discussion of the discrete state-space representation of the problem and the model inaccuracies due to mismodeling effects. We present two sequential inference algorithms: (i) a method based on particle filtering (PF) to estimate the time-evolving states of a neuron under the assumption of perfect model knowledge and (ii) an enhanced version where model parameters are jointly estimated, and thus, the rather strong assumption of perfect model knowledge is relaxed. We provide exhaustive computer simulation results to validate the algorithms and observe that they are attaining the theoretical lower bounds of accuracy, which are derived in the paper as well.

In this paper, we use the powerful tools of PF to make inferences in general state-space models. PF are a set of methods able to sample from the marginal filtering distribution in situations where analytical solutions are hard to work out or simply impossible. In the recent years, PFs played an important role in many research areas such as signal detection and demodulation, target tracking, positioning, Bayesian inference, audio processing, financial modeling, computer vision, robotics, control, or biology [18–24]. At a glance, PF approximates the filtering distribution of states given measurements by a set of random points, properly weighted according to Bayes' rule. The generation of the random particles can be done through a variety of distributions, known as importance density. Particularly, we formulate the problem at hand and observe that it is possible to use the optimal importance density [25]. This distribution generates particles close to the target distribution, and thus, it can be shown to reduce the variance of the particles. As a consequence, for a fix number of particles, usage of this approach (not always possible) leads to better accuracy results than other choices. To the authors' knowledge, the utilization of such sampling distribution is novel in the context of neural model filtering. Similar works have used PF to track neural dynamics, but with no optimal importance density (see [2, 3, 26]), or to estimate synaptic input from sub-threshold recordings [11], as opposite to our proposed approach where we aim at providing estimates during the, highly nonlinear, spike regime. These references use the expectation-maximization algorithm to estimate the model parameters. In this paper, we use the Particle Marginal Markov-Chain Monte-Carlo (PMCMC) method to estimate the parameters. Lighter filtering methods based on the Gaussian assumption were considered in the literature (see [12, 13, 27] for instance), but the assumption might not hold in general, for instance, due to outliers in the membrane measurements or if more sophisticated models for the synaptic conductances are considered. In these situations, a PF approach seems more appropriate. As mentioned earlier, the focus here is on highly efficient and reliable filtering methods, rather than on computationally light inference methods. Summing up,

our method jointly treats the features of handling single-voltage traces governed by nonlinear models, estimating both neuron parameters and synaptic conductances, even in the spiking regimes, and using optimal importance density for the PF together with a MCMC algorithm. The above references cope with some of these features, but to our knowledge, none of the recent methods in the literature accounts for all of them. It is worth noting that other simulation-based solutions can be adopted besides the PMCMC. For instance, the works [28–32] tackle state estimation and model fitting problems jointly.

The remainder of the article is organized as follows. In Section 2, we expose the problem and present the statistical model, essentially a discretization of the well-known Morris-Lecar model, and we analyze the model inaccuracies as well. Next, in Section 3, we present the different inference algorithms we apply depending on the knowledge of the system. Results are given in Section 4, where we tackle three inference problems: (i) when the parameters defining the model are known; (ii) when the parameters of the model are unknown, and thus, they need to be estimated; and (iii) estimation of synaptic conductances from voltage traces assuming unknown model parameters. Finally, Section 5 concludes the paper with final remarks.

2 Problem statement and model
2.1 Measurement modeling
The recording of the membrane potential is a physical process, including some approximations/inaccuracies involving:

1. Voltage observations are noisy. This is due, in part, to the thermal noise at the sensing device, non-ideal conditions in experimental setups, etc.
2. Recorded observations are discrete. All sensing devices record data by sampling at discrete time-instants k, at a sampling frequency $f_s = 1/T_s$, the continuous-time natural phenomena. This is the task of an Analog-to-Digital Converter (ADC). Moreover, these samples are typically digitized, i.e., expressed by a finite number of bits. This latter issue is not tackled in the work as we assume that modern computer capabilities allow us to sample with relatively large number of bits per sample.

The problem can thus be posed in the form of a discrete-time, state-space model, where the observations are

$$y_k \sim \mathcal{N}\left(\nu_k, \sigma_{y,k}^2\right), \tag{1}$$

with ν_k representing the nominal membrane potential and $\sigma_{y,k}^2$ modeling the noise variance due to the sensor or the instrumentation inaccuracies when performing the experiment. To provide comparable results, we define the

signal-to-noise ratio (SNR) as $\text{SNR} = P_s/P_n$, with P_s being the average signal power and $P_n = \sigma_{y,k}^2$ the noise power.

2.2 Neural dynamical system

The methods presented in this paper rely on continuous models for the evolution of the voltage-traces and the hidden variables of a neuron. Our reference framework are conductance-based models endowed with a synaptic input term I_{syn} and an (steady) applied current I_{app}, that is, equations of type

$$C_m \dot{v} = -\bar{g}_L (v - E_L) - \sum_{j \in \mathcal{J}} I_j - I_{\text{syn}} + I_{\text{app}}, \qquad (2)$$

where $\bar{g}_L (v - E_L)$ is the leakage current and each $I_j = \bar{g}_j \, p_j (v - E_j)$ is the time-varying ionic current for the jth ionic species, $\mathcal{J} = \{\text{Na}, \text{K}, \text{Cl}, \text{Ca}, \dots\}$, where $\bar{g}_j$ is the maximal conductance, p_j involves the so-called gating variables, and E_j is the reversal potential of the ionic channel. The synaptic input is expressed as $I_{\text{syn}} = g_E(t)(v(t) - E_E) + g_I(t)(v(t) - E_I)$. We use the so-called *effective point-conductance* model, see [7, 33], where the excitatory/inhibitory global conductances are treated as Ornstein-Uhlenbeck (OU) processes

$$\dot{g}_u(t) = -\frac{1}{\tau_u} \left(g_u(t) - g_{u,0} \right) + \sqrt{\frac{2\sigma_u^2}{\tau_u}} \chi(t) \qquad (3)$$

where $u = \{E, I\}$ and $\chi(t)$ is a zero-mean, white noise, Gaussian process with unit variance. Then, the OU process has mean $g_{u,0}$, standard deviation σ_u, and time constant τ_u. This simple model was shown in [33] to yield a valid description of the synaptic noise, capturing the properties of more complex models. Other dynamics could be considered instead.

Concerning the neuron model, that is, the equation $C_m \dot{v} = -\bar{g}_L (v - E_L) - \sum_{j \in \mathcal{J}} I_j$, in this work, we focus on the Morris-Lecar model [34] for the sake of clarity, which is able to model a wide variety of neural dynamics; see [35]. Details of the Morris-Lecar model can be found in Appendix 1. The unknown state vector in this case is composed of the membrane potential, v, and a unique ionic (K^+-)current involving the gating variable n. We write

$$\boldsymbol{x}_k = \begin{pmatrix} v_k \\ n_k \end{pmatrix} . \qquad (4)$$

Notice that the Morris-Lecar neuron model is defined by a system of continuous-time, ordinary differential equations (ODEs) of the form $\dot{\boldsymbol{x}} = f(\boldsymbol{x})$. In general, mathematical models for neurons are of this type. However, due to the sampled recording of measurements, we are interested in expressing the model in the form of a discrete state-space,

$$\boldsymbol{x}_k = f_k(\boldsymbol{x}_{k-1}) + \boldsymbol{v}_k \qquad (5)$$

where $\boldsymbol{v}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_{x,k})$ is the process noise which includes the model inaccuracies. The covariance matrix $\boldsymbol{\Sigma}_{x,k}$ is used to quantify our confidence in the model that maps $f_k : \{v_{k-1}, n_{k-1}\} \mapsto \{v_k, n_k\}$. In general, obtaining a closed-form analytical expression for f_k without approximations is not possible. In such a case, we could use the Euler method:

$$\dot{\boldsymbol{x}} \doteq \frac{d\boldsymbol{x}}{dt} \approx \frac{\Delta \boldsymbol{x}}{\Delta t} = \frac{\boldsymbol{x}(t + \text{T}_s) - \boldsymbol{x}(t)}{\text{T}_s} = f(\boldsymbol{x}(t)), \qquad (6)$$

where $\Delta t = \text{T}_s$ is the sampling period. Thus, we can write (5) as

$$\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \text{T}_s f(\boldsymbol{x}_{k-1}) , \qquad (7)$$

which is of the Markovian type.

If we focus on the Morris-Lecar model, the resulting discrete version of the ODE system in (40)–(41) is:

$$\begin{aligned} v_k = v_{k-1} - \frac{\text{T}_s}{C_m} \Big(& \bar{g}_L(v_{k-1} - E_L) \\ & + \bar{g}_{\text{Ca}} m_\infty(v_{k-1})(v_{k-1} - E_{\text{Ca}}) \\ & + \bar{g}_K n_{k-1} (v_{k-1} - E_K) - I_{\text{app}} \Big) \end{aligned} \qquad (8)$$

$$n_k = n_{k-1} + \text{T}_s \phi \frac{n_\infty(v_{k-1}) - n_{k-1}}{\tau_n(v_{k-1})} , \qquad (9)$$

with $m_\infty(v_k)$, $n_\infty(v_k)$ and $\tau_n(v_k)$ defined in Appendix 1. Then, (8) and (9) can be interpreted as $\boldsymbol{x}_k = f_k(\boldsymbol{x}_{k-1})$.

2.3 State-space formulation

The goal is to express the inference problem in state-space formulation and apply stochastic filtering tools learned from signal processing. The final ingredient to do so is to introduce the so-called process noise in the state equation. Leveraging on the observation equation in (1), the state-space can be formulated as

$$\boldsymbol{x}_k = f_k(\boldsymbol{x}_{k-1}) + \begin{pmatrix} v_{v,k} \\ v_{n,k} \end{pmatrix} \qquad (10)$$

$$y_k = v_k + v_{y,k} , \qquad (11)$$

where the noise terms $v_{v,k}$ and $v_{n,k}$ are assumed jointly Gaussian with covariance matrix $\boldsymbol{\Sigma}_{x,k}$. Further details of this matrix are discussed in Section 2.4. The measurement noise $v_{y,k}$ is assumed zero-mean, Gaussian, and with variance $\sigma_{y,k}^2$. Notice that the system is characterized by Gaussian distributions and is linear in the observations; this allows for an optimal design of the proposal density in the PF as exploited in Section 3.1.

This general form of Markovian type is preserved when the model is extended with a couple of OU processes associated to the excitatory and inhibitory synaptic conductances. In this case, the resulting state-space model is composed by the Morris-Lecar model used so far (with the peculiarity that the term $-I_{\text{syn}}$ is added to (40)), plus

the OU stochastic process in (3) describing I_{syn}. Therefore, the continuous-time state is $\boldsymbol{x} = (v, n, g_{\mathrm{E}}, g_{\mathrm{I}})^{\top}$. The discrete version of the state-space is used again.

2.4 Model inaccuracies

The proposed estimation method relies on the fact that the neuron model is known. This is true to some extent, but most of the parameters in the Morris-Lecar model discussed are to be estimated. Typically, this model calibration is done beforehand, but as we will see later in Section 3.2, this could be done in parallel to the filtering process. Therefore, the robustness of the method to possible inaccuracies should be assessed. In this section, we point out possible causes of mismodeling. Computer simulations are later used to characterize the performance of the proposed methods under these impairments.

In the single-neuron model considered, three major sources of inaccuracies can be identified. First, the applied current I_{app} can be itself noisy, with a variance depending on the quality of the instrumentation used and the experiment itself. We model the actual applied current as a random variable

$$I_{\mathrm{app}} = I_o + \nu_{I,k} \,, \quad \nu_{I,k} \sim \mathcal{N}\left(0, \sigma_I^2\right) \,, \tag{12}$$

where I_o is the nominal current applied and σ_I^2 the variance around this value. Plugging (12) into (8), we obtain that the contribution of I_{app} to the noise term is $\frac{\mathrm{T}_s}{C_m} \nu_{I,k} \sim \mathcal{N}\left(0, (\mathrm{T}_s/C_m)^2 \sigma_I^2\right)$.

Secondly, when the conductance of the leakage term is estimated beforehand, some inaccuracies might be taken into account. In general, this term is considered constant in the models although it gathers relatively distinct phenomena that can potentially be time-varying. The maximal conductance of the leakage term is therefore inaccurate and modeled as

$$\bar{g}_L = \bar{g}_L^o + \nu_{g,k} \,, \quad \nu_{g,k} \sim \mathcal{N}\left(0, \sigma_g^2\right) \,, \tag{13}$$

where $\bar{g}_L^o$ is the nominal, estimated conductance and σ_g^2 the variance of this estimate. Similarly, inserting (13) into (8), we see that the contribution of $\bar{g}_L$ to the noise term is $\frac{\mathrm{T}_s}{C_m} \nu_{g,k} \sim \mathcal{N}\left(0, (\mathrm{T}_s/C_m)^2 \left(\nu_{k-1} - E_{\mathrm{L}}\right) \sigma_g^2\right)$.

Finally, the parameters in $m_\infty(\nu_k)$, $n_\infty(\nu_k)$, and $\tau_n(\nu_k)$ are to be estimated. In general, these parameters can be properly obtained off-line by standard methods; see [36]. However, as they are estimates, a residual error typically remains. To account for these inaccuracies, we consider that the equation governing the evolution of gating variables is corrupted by a zero-mean additive white Gaussian process with variance σ_n^2.

This analysis allows us to construct a realistic $\boldsymbol{\Sigma}_{x,k}$, as the contribution of the aforementioned inaccuracies. In a practical setup, in order to compute the noise variance due to leakage, we need to use the approximation $\hat{\nu}_{k-1} \approx$

ν_{k-1}, where $\hat{\nu}_{k-1}$ is estimated by the filtering method. We construct the covariance matrix of the model as

$$\boldsymbol{\Sigma}_{x,k} = \begin{pmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_n^2 \end{pmatrix} \,, \tag{14}$$

where we used that the overall noise in the voltage model is $\frac{\mathrm{T}_s}{C_m} \left(\nu_{I,k} - \nu_{g,k}\right) \sim \mathcal{N}\left(0, \sigma_v^2\right)$ and

$$\sigma_v^2 = \left(\frac{\mathrm{T}_s}{C_m}\right)^2 \left(\sigma_I^2 + \left(\hat{\nu}_{k-1} - E_{\mathrm{L}}\right)^2 \sigma_g^2\right) \tag{15}$$

as an estimate of σ_v^2, provided accurate knowledge of σ_I^2 and σ_g^2. Otherwise, the covariance matrix of the process could be estimated by other means, as the ones presented in Section 3.2 for mixed state-parameter estimation in nonlinear filtering problems.

Notice that we are implicitly assuming white processes due to the diagonal structure of $\Sigma_{x,k}$. It is worth mentioning that, should correlated noise be a more realistic model, the method proposed in this article would be still valid. The proposed method can cope with colored noise statistics since $\Sigma_{x,k}$ can be used seamlessly if it is not diagonal.

3 Methods

Two filtering methods are proposed here, depending on the knowledge regarding the underlying dynamical model. Section 3.1 presents an algorithm able to estimate the states in $\boldsymbol{x}_k$ by a PF methodology, the particularity being that an optimal distribution is used to draw the random samples characterizing the joint filtering distribution of interest. This method assumes knowledge of the parameter values of the system model, although we account for some inaccuracies as detailed in Section 2.4. An enhanced version of this method is presented in Section 3.2, where we relax the assumption of knowing the parameter values. Leveraging on a PMCMC algorithm and the use of the optimal importance density as in the first method, we present a method that is able to filter $\boldsymbol{x}_k$ while estimating the values describing the neuron model.

3.1 Sequential estimation of voltage traces and gating variables

The type of problems we are interested in involve the estimation of time-evolving signals that can be expressed through a state-space formulation. Particularly, estimation of the states in a single-neuron model from noisy voltage traces can be readily seen as a filtering problem. Bayesian theory provides the mathematical tools to deal with such problems in a systematic manner. The focus is on sequential methods that can incorporate new available measurements as they are recorded without the need for reprocessing all past data.

Bayesian filtering involves the recursive estimation of states $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$ given measurements $y_k \in \mathbb{R}$ at time t

based on all available measurements, $y_{1:k} = \{y_1, \ldots, y_k\}$. To that aim, we are interested in the filtering distribution $p(\boldsymbol{x}_k|y_{1:k})$. Assuming the Markovian property in (10) and (11), the distribution can be recursively expressed as

$$p\left(\boldsymbol{x}_k|y_{1:k}\right) = \frac{p\left(y_k|\boldsymbol{x}_k\right) p\left(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}\right)}{p\left(y_k|y_{1:k-1}\right)} p\left(\boldsymbol{x}_{k-1}|y_{1:k-1}\right) ,$$

(16)

with $p(y_k|\boldsymbol{x}_k)$ and $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ referred to as the likelihood and the prior distributions, respectively. Unfortunately, (16) can only be obtained in closed-form in some special cases, and in more general setups, we should resort to more sophisticated methods. In this paper, we consider PF to cope with the nonlinearity of the model. Although other lighter approaches might be possible as well [22], we seek the maximum accuracy regardless the involved computational cost. Theoretically, for sufficiently large number of particles, particle filters offer such performances.

Particle filters, see [18, 20, 21, 23], approximate the filtering distribution by a set of N weighted random samples, forming the random measure $\left\{\boldsymbol{x}_k^{(i)}, w_k^{(i)}\right\}_{i=1}^N$. These random samples are drawn from the importance density distribution, $\pi(\cdot)$,

$$\boldsymbol{x}_k^{(i)} \sim \pi\left(\boldsymbol{x}_k|\boldsymbol{x}_{0:k-1}^{(i)}, y_{1:k}\right)$$

(17)

and weighted according to the general formulation

$$w_k^{(i)} \propto w_{k-1}^{(i)} \frac{p\left(y_k|\boldsymbol{x}_{0:k}^{(i)}, y_{1:k-1}\right) p\left(\boldsymbol{x}_k^{(i)}|\boldsymbol{x}_{k-1}^{(i)}\right)}{\pi\left(\boldsymbol{x}_k^{(i)}|\boldsymbol{x}_{0:k-1}^{(i)}, y_{1:k}\right)} .$$

(18)

The importance density from which particles are drawn is a key issue in designing efficient PFs. It is well known that the optimal importance density is

$$\pi\left(\boldsymbol{x}_k|\boldsymbol{x}_{0:k-1}^{(i)}, y_{1:k}\right) = p\left(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{(i)}, y_k\right),$$

in the sense that it minimizes the variance of importance weights. Weights are then computed using (18) as $w_k^{(i)} \propto w_{k-1}^{(i)} p\left(y_k|\boldsymbol{x}_{k-1}^{(i)}\right)$. This choice requires the ability to draw from $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{(i)}, y_k)$ and to evaluate $p\left(y_k|\boldsymbol{x}_{k-1}^{(i)}\right)$. In general, the two requirements cannot be met, and one needs to resort to suboptimal choices. However, given the particular structure of the state-space model, we are able to use the optimal importance density. The fact that the model is Gaussian and the observations are related linearly to states allow to solve for the conditional distribution of states from the joint Gaussian distribution of states and observations. The equations are:

$$p\left(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}^{(i)}, y_k\right) = \mathcal{N}\left(\boldsymbol{\mu}_{\pi,k}^{(i)}, \boldsymbol{\Sigma}_{\pi,k}\right)$$

(19)

with

$$\boldsymbol{\mu}_{\pi,k}^{(i)} = \boldsymbol{\Sigma}_{\pi,k}\left(\boldsymbol{\Sigma}_{x,k}^{-1} f_k\left(\boldsymbol{x}_{k-1}^{(i)}\right) + \frac{y_k}{\sigma_{y,k}^2}\right)$$

(20)

$$\boldsymbol{\Sigma}_{\pi,k} = \left(\boldsymbol{\Sigma}_{x,k}^{-1} + \sigma_{y,k}^{-2}\mathbf{I}\right)^{-1} ,$$

(21)

and the importance weights can be updated using

$$p\left(y_k|\boldsymbol{x}_{k-1}^{(i)}\right) = \mathcal{N}\left(\mathbf{h}^\top f_k\left(\boldsymbol{x}_{k-1}^{(i)}\right), \mathbf{h}^\top \boldsymbol{\Sigma}_{x,k}\mathbf{h} + \sigma_{y,k}^2\right) ,$$

(22)

with $\mathbf{h} = (1,0)^\top$. The PF provides a discrete approximation of the filtering distribution of the form

$$p(\boldsymbol{x}_k|y_{1:k}) \approx \sum_{i=1}^N w_k^{(i)} \delta\left(\boldsymbol{x}_k - \boldsymbol{x}_k^{(i)}\right),$$

which gathers all information from $\boldsymbol{x}_k$ that the measurements up to time k provide. The minimum mean square error estimator can be obtained as

$$\hat{\boldsymbol{x}}_k = \sum_{i=1}^N w_k^{(i)} \boldsymbol{x}_k^{(i)} ,$$

(23)

where $\hat{\boldsymbol{x}}_k = \left(\hat{v}_k, \hat{n}_k\right)^\top$. Recall that the method discussed in this section could be easily adapted to other neuron models by simply substituting the corresponding transition function f_k and constructing the state vector $\boldsymbol{x}_k$ accordingly.

As a final step, PFs incorporate a resampling strategy to avoid the collapse of particles into a single state point. Resampling consists in eliminating particles with low importance weights and replicating those in high-probability regions [37]. The overall algorithm can be looked up in Algorithm 1 at instance k. Notice that this version of the algorithm requires knowledge of noise statistics and all the model parameters, which for the Morris-Lecar model are

$$\boldsymbol{\Theta} = \left(\bar{g}_L, E_L, \bar{g}_{Ca}, E_{Ca}, \bar{g}_K, E_K, \phi, V_1, V_2, V_3, V_4\right)^\top . \quad (24)$$

When we add the dynamics of the synaptic conductances, the vector $\boldsymbol{\Theta}$ of model parameters also includes τ_E, τ_I, $g_{E,0}$, $g_{I,0}$, σ_E, and σ_I.

3.2 Joint estimation of states and model parameters

In practice, the parameters in (24) might not be known. It is reasonable to assume that $\boldsymbol{\Theta}$, or a subset of these parameters $\theta \subseteq \boldsymbol{\Theta}$, are unknown and need to be estimated at the same time the filtering method in Algorithm 1 is executed. Therefore, the ultimate goal in this case is to estimate jointly the time evolving states and the unknown parameters of the model, $\boldsymbol{x}_{1:k}$ and θ, respectively.

Joint estimation of states and parameters is a longstanding problem in Bayesian filtering and specially hard to

---

Algorithm 1 Particle filtering with optimal importance density

---

Require: N, $\boldsymbol{\Sigma}_{x,k}$, $\sigma_{y,k}^2$, $\boldsymbol{\Theta}$, $\left\{ \boldsymbol{x}_{k-1}^{(i)}, w_{k-1}^{(i)} \right\}_{i=1}^{N}$ and y_k

Ensure: $\left\{ \boldsymbol{x}_k^{(i)}, w_k^{(i)} \right\}_{i=1}^{N}$ and $\hat{\boldsymbol{x}}_k$

1: Calculate $\boldsymbol{\Sigma}_{\pi,k} = \left(\boldsymbol{\Sigma}_{x,k}^{-1} + \sigma_{y,k}^{-2} \mathbf{I} \right)^{-1}$

2: **for** $i = 1$ to N **do**

3:　　　Calculate $\boldsymbol{\mu}_{\pi,k}^{(i)} = \boldsymbol{\Sigma}_{\pi,k} \left(\boldsymbol{\Sigma}_{x,k}^{-1} f_k \left(\boldsymbol{x}_{k-1}^{(i)} \right) + \frac{y_k}{\sigma_{y,k}^2} \right)$

4:　　　Generate $\boldsymbol{x}_k^{(i)} \sim \mathcal{N}(\boldsymbol{\mu}_{\pi,k}^{(i)}, \boldsymbol{\Sigma}_{\pi,k})$

5:　　　Calculate $\tilde{w}_k^{(i)} = w_{k-1}^{(i)} \dfrac{p\left(y_k | \boldsymbol{x}_{0:k}^{(i)}, y_{1:k-1}\right) p\left(\boldsymbol{x}_k^{(i)} | \mathbf{x}_{k-1}^{(i)}\right)}{\mathcal{N}(\boldsymbol{\mu}_{\pi,k}^{(i)}, \boldsymbol{\Sigma}_{\pi,k})}$

6: **end for**

7: **for** $i = 1$ to N **do**

8:　　　Normalize weights: $w_k^{(i)} = \dfrac{\tilde{w}_k^{(i)}}{\sum_{j=1}^{N} \tilde{w}_k^{(j)}}$

9: **end for**

10: MMSE state estimation: $\hat{\boldsymbol{x}}_k = \sum_{i=1}^{N} w_k^{(i)} \boldsymbol{x}_k^{(i)}$

11: $\left\{ \boldsymbol{x}_k^{(i)}, 1/N \right\}_{i=1}^{N} = \mathrm{Resample}\left(\left\{ \boldsymbol{x}_k^{(i)}, w_k^{(i)} \right\}_{i=1}^{N} \right)$

---

handle in the context of PFs. Refer to [38–40] and their references for a complete survey. Here, we follow the approach in [41] and make use of the so-called PMCMC to enhance the presented PF algorithm with parameter estimation capabilities. In the remainder of this section, we provide the basic ideas to use the algorithm, following a similar approach as in [24]. Connections to other methods are discussed in [42].

Following the Bayesian philosophy we adopt here, the problem fundamentally reduces to assigning an a priori distribution for the unknown parameter $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$ and extending the state-space model (here, we adopt its probabilistic representation)

$$\boldsymbol{\theta} \sim p(\boldsymbol{\theta}) \tag{25}$$

$$\boldsymbol{x}_k \sim p(\boldsymbol{x}_k | \boldsymbol{x}_{k-1}, \boldsymbol{\theta}) \quad \text{for } k \geq 1 \tag{26}$$

$$y_k \sim p(y_k | \boldsymbol{x}_k, \boldsymbol{\theta}) \quad \text{for } k \geq 1 \tag{27}$$

and initial state distribution $\boldsymbol{x}_0 \sim p(\boldsymbol{x}_0 | \boldsymbol{\theta})$. Applying Bayes' rule, the full posterior distribution can be expressed as

$$p\left(\boldsymbol{x}_{0:T}, \boldsymbol{\theta} | y_{1:T}\right) = \frac{p(y_{1:T} | \boldsymbol{x}_{0:T}, \boldsymbol{\theta}) p\left(\boldsymbol{x}_{0:T} | \boldsymbol{\theta}\right) p(\boldsymbol{\theta})}{p(y_{1:T})} \tag{28}$$

with

$$p\left(y_{1:T} | \boldsymbol{x}_{0:T}, \boldsymbol{\theta}\right) = \prod_{k=1}^{T} p\left(y_k | \boldsymbol{x}_k, \boldsymbol{\theta}\right) \tag{29}$$

$$p\left(\boldsymbol{x}_{0:T} | \boldsymbol{\theta}\right) = p\left(\boldsymbol{x}_0 | \boldsymbol{\theta}\right) \prod_{k=1}^{T} p\left(\boldsymbol{x}_k | \boldsymbol{x}_{k-1}, \boldsymbol{\theta}\right) . \tag{30}$$

Notice here that we are dealing with a finite horizon of observations T. Then, from a Bayesian perspective, the estimation of $\boldsymbol{\theta}$ is equivalent to obtaining its marginal posterior distribution $p(\boldsymbol{\theta} | y_{1:T}) = \int p(\boldsymbol{x}_{0:T}, \boldsymbol{\theta} | y_{1:T}) d\boldsymbol{x}_{0:T}$. However, this is in general extremely hard to compute analytically, and one needs to find work-arounds. Evaluation of the full posterior turns out to be not only computationally prohibitive but useless if states cannot be marginalized out analytically. Alternative methods resort on the factorization of the parameter marginal distribution as $p(\boldsymbol{\theta} | y_{1:T}) \propto p(y_{1:T} | \boldsymbol{\theta}) p(\boldsymbol{\theta})$ and how Bayesian filters can be transformed to provide characterizations of the marginal likelihood distribution $p(y_{1:T} | \boldsymbol{\theta})$ and related quantities. The marginal likelihood distribution can be recursively factorized in terms of the predictive distributions of the observations:

$$p(y_{1:T} | \boldsymbol{\theta}) = \prod_{k=1}^{T} p(y_k | y_{1:k-1}, \boldsymbol{\theta}),$$

with $p(y_k | y_{1:k-1}, \boldsymbol{\theta}) = \int p(y_k | \boldsymbol{x}_k, \boldsymbol{\theta}) p(\boldsymbol{x}_k | y_{1:k-1}, \boldsymbol{\theta}) d\boldsymbol{x}_k$ obtained straightforwardly as a byproduct of any of the Bayesian filtering methods; see [22].

A useful transformation of the marginal likelihood is the so-called *energy function*, which is sometimes more convenient to deal with. The energy function is defined as $\varphi_T(\boldsymbol{\theta}) = -\ln p(y_{1:T} | \boldsymbol{\theta}) - \ln p(\boldsymbol{\theta})$ or, equivalently, $p(\boldsymbol{\theta} | y_{1:T}) \propto \exp(-\varphi_T(\boldsymbol{\theta}))$. The energy function can then be recursively computed as a function of the predictive distribution

$$\varphi_0(\boldsymbol{\theta}) = -\ln p(\boldsymbol{\theta}) \tag{31}$$

$$\varphi_k(\boldsymbol{\theta}) = \varphi_{k-1}(\boldsymbol{\theta}) - \ln p(y_k | y_{1:k-1}, \boldsymbol{\theta}) \tag{32}$$

for $k \geq 1$.

Then, the basic problem is to obtain an estimate of the predictive distribution $p(y_k | y_{1:k-1}, \boldsymbol{\theta})$ from the PF we have designed in Section 3.1 and use it in conjunction with $p(\boldsymbol{\theta})$ to infer the marginal distribution $p(\boldsymbol{\theta} | y_{1:T})$ of interest. This latter step can be performed in several ways, from which we choose to use the Markov-Chain Monte-Carlo (MCMC) methodology to continue with a fully Bayesian solution. Besides, it is known to be the solution that provides the best results when used in a PF. Next, we detail how $\varphi_k(\boldsymbol{\theta})$ can be obtained from a PF algorithm, we present the MCMC method for parameter estimation, and finally we sketch the overall algorithm.

3.2.1 Computing the energy function from particle filters

The modification needed is rather small. Actually, it is non-invasive in the sense that the algorithm remains the same and the energy function can be computed adding some extra formulae. Recall that the predictive distribution $p(y_k | y_{1:k-1}, \boldsymbol{\theta})$ is composed of two distributions and

that the PF provides characterizations of these two distributions. Then, one could use a particle approximation $p(y_k|y_{1:k-1}, \boldsymbol{\theta}) \approx \hat{p}(y_k|y_{1:k-1}, \boldsymbol{\theta}) = \sum_{i=1}^{N} w_{k-1}^{(i)} \zeta_k^{(i)}$ with $w_{k-1}^{(i)}$ as in the original algorithm and

$$\zeta_k^{(i)} = \frac{p\left(y_k|\boldsymbol{x}_k^{(i)}, \boldsymbol{\theta}\right) p\left(\boldsymbol{x}_k^{(i)}|\boldsymbol{x}_{k-1}^{(i)}, \boldsymbol{\theta}\right)}{\pi\left(\boldsymbol{x}_k^{(i)}|\boldsymbol{x}_{0:k-1}^{(i)}, y_{1:k}, \boldsymbol{\theta}\right)} . \tag{33}$$

Then, it is straightforward to identify the energy function approximation as

$$\varphi_T(\boldsymbol{\theta}) \approx -\ln p(\boldsymbol{\theta}) - \sum_{k=1}^{T} \ln \hat{p}(y_k|y_{1:k-1}, \boldsymbol{\theta}) \tag{34}$$

$$= -\ln p(\boldsymbol{\theta}) - \sum_{k=1}^{T} \ln \sum_{i=1}^{N} w_{k-1}^{(i)} \zeta_k^{(i)} \tag{35}$$

which can be computed recursively in the PF algorithm and becomes an approximation $\hat{\varphi}_T(\boldsymbol{\theta})$ of the energy function.

3.2.2 The Particle Markov-Chain Monte-Carlo algorithm

Once an approximation of the energy function is available, we can apply MCMC to infer the marginal distribution of the parameters. MCMC methods constitute a general methodology to generate samples recursively from a given distribution by randomly simulating from a Markov chain [43–47]. There are many algorithms implementing the MCMC concept, the Metropolis-Hastings (MH) algorithm being one of the most popular. At the jth iteration, the MH algorithm samples a candidate point $\boldsymbol{\theta}^*$ from a proposal distribution $q(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(j-1)})$ based on the previous sample $\boldsymbol{\theta}^{(j-1)}$. Starting from an arbitrary value $\boldsymbol{\theta}^{(0)}$, the MH algorithm accepts the new candidate point (meaning that it was generated from the target distribution, $p(\boldsymbol{\theta}|y_{1:T})$) using the rule

$$\boldsymbol{\theta}^{(j)} = \begin{cases} \boldsymbol{\theta}^*, & \text{if } u \leq \alpha^{(j)} \\ \boldsymbol{\theta}^{(j)}, & \text{otherwise} \end{cases} \tag{36}$$

where u is drawn randomly from a uniform distribution, $u \sim \mathcal{U}(0, 1)$, and

$$\alpha^{(j)} = \min\left\{1, \exp\left(\varphi_T(\boldsymbol{\theta}^{(j-1)}) - \varphi_T(\boldsymbol{\theta}^*)\right) \frac{q\left(\boldsymbol{\theta}^{(j-1)}|\boldsymbol{\theta}^*\right)}{q\left(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(j-1)}\right)}\right\}$$

is referred to as the acceptance probability.

It is critical for the performance of the algorithm the choice of the proposal density. A common choice is $q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(j-1)}) = \mathcal{N}(\boldsymbol{\theta}; \boldsymbol{\theta}^{(j-1)}, \boldsymbol{\Sigma}^{(j-1)})$ with the selection of the transitional covariance remaining as the tuning $\boldsymbol{\Sigma}^{(j-1)}$ parameter. This covariance can be adapted as iterations of the MCMC method progress. In this work, we have adopted the Robust Adaptive Metropolis (RAM) algorithm [48], although other methods could be considered

for the same purpose [49, 50]. The RAM algorithm is provided in Algorithm 2. We use the notation that $\mathbf{S} = \text{Chol}(\mathbf{A})$ denotes the Cholesky factorization of an Hermitian positive-definite matrix $\mathbf{A}$ such that $\mathbf{A} = \mathbf{S}\mathbf{S}^\top$, and $\mathbf{S}$ is a lower triangular matrix [51]. The RAM algorithm outputs a set of samples $\left\{\boldsymbol{\theta}^{(j)}\right\}_{j=1}^{M}$, where M is the number of iterations of th e MCMC procedure. These samples are originated from the target distribution $\left\{\boldsymbol{\theta}^{(j)} \sim p(\boldsymbol{\theta}|y_{1:T})\right\}_{j=1}^{M}$, which can be used to approximate (after neglecting the first samples corresponding to the transient phase of the algorithm) it as

$$p(\boldsymbol{\theta}|y_{1:T}) \approx \frac{1}{M} \sum_{j=1}^{M} \delta\left(\boldsymbol{\theta} - \boldsymbol{\theta}^{(j)}\right), \tag{37}$$

and one can obtain the desired statistics from the characterization of the marginal distribution. For instance, point estimates of the parameter like

$$\hat{\boldsymbol{\theta}}^{\text{MMSE}} = \frac{1}{M} \sum_{j=1}^{M} \boldsymbol{\theta}^{(j)} \qquad \text{or} \qquad \hat{\boldsymbol{\theta}} = \boldsymbol{\theta}^{(M)} . \tag{38}$$

The main assumption in Algorithm 2 is the ability of evaluating the energy function, $\varphi_T(\cdot)$. We have seen earlier how this can be done in a PF. Roughly speaking, the PMCMC algorithm consists of putting together these two algorithms [41]. We refer to Algorithm 3 for the resulting PMCMC method.

Algorithm 2 Robust Adaptive Metropolis

Require: $M, \boldsymbol{\theta}^{(0)}, \boldsymbol{\Sigma}^{(0)}, \gamma \in (\frac{1}{2}, 1], \bar{\alpha}_*$, and $\varphi_T(\cdot)$

Ensure: $\left\{\boldsymbol{\theta}^{(j)}\right\}_{j=1}^{M}$

1: Initialize: $\mathbf{S}_0 = \text{Chol}\left(\boldsymbol{\Sigma}^{(0)}\right)$ and $\varphi_T\left(\boldsymbol{\theta}^{(0)}\right) = 0$

2: **for** $j = 1$ to M **do**

3:　　Draw $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

4:　　Compute $\boldsymbol{\theta}^* = \boldsymbol{\theta}^{(j-1)} + \mathbf{S}_{j-1}\mathbf{a}$

5:　　Compute $\alpha^{(j)} = \min\left\{1, \exp\left(\varphi_T\left(\boldsymbol{\theta}^{(j-1)}\right) - \varphi_T(\boldsymbol{\theta}^*)\right)\right\}$

6:　　Draw $u \sim \mathcal{U}(0, 1)$

7:　　**if** $u \leq \alpha^{(j)}$ **then**

8:　　　$\boldsymbol{\theta}^{(j)} = \boldsymbol{\theta}^*$

9:　　**else**

10:　　　$\boldsymbol{\theta}^{(j)} = \boldsymbol{\theta}^{(j-1)}$

11:　　**end if**

12:　　Compute $\eta^{(j)} = j^{-\gamma}$

13:　　Compute $\mathbf{D}_j = \left(\mathbf{I} + \eta^{(j)}\left(\alpha^{(j)} - \bar{\alpha}_*\right)\frac{\mathbf{a}\mathbf{a}^\top}{||\mathbf{a}||^2}\right)$

14:　　Compute $\mathbf{S}_j = \text{Chol}\left(\mathbf{S}_{j-1}\mathbf{D}_j\mathbf{S}_{j-1}^\top\right)$

15: **end for**

Algorithm 3 Joint state-parameter estimation by Particle Marginal MCMC

Require: $y_{1:T}$, M, $\boldsymbol{\theta}^{(0)}$, $\boldsymbol{\Sigma}^{(0)}$, $\gamma \in (1/2, 1]$, $\bar{\alpha}_*$, and $\varphi_T(\cdot)$
Ensure: $\hat{\boldsymbol{x}}_{1:T}$ and $\hat{\boldsymbol{\theta}}$

1: Initialize: $\mathbf{S}_0 = \text{Chol}\left(\boldsymbol{\Sigma}^{(0)}\right)$ and $\hat{\varphi}_T(\boldsymbol{\theta}^{(0)}) = 0$
2: **for** $j = 1$ to M **do**
3:      Draw $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
4:      Compute $\boldsymbol{\theta}^* = \boldsymbol{\theta}^{(j-1)} + \mathbf{S}_{j-1}\mathbf{a}$
5:      Run the PF in Algorithm 1 with model parameters set to $\boldsymbol{\theta}^*$.
        Required outputs:
           State filtering $\hat{\boldsymbol{x}}_{1:T}^*$ as in (23)
           Energy function $\hat{\varphi}_T(\boldsymbol{\theta}^*)$ as in (35)
6:      Compute $\alpha^{(j)} = \min\left\{1, \exp\left(\hat{\varphi}_T\left(\boldsymbol{\theta}^{(j-1)}\right) - \hat{\varphi}_T(\boldsymbol{\theta}^*)\right)\right\}$
7:      Draw $u \sim \mathcal{U}(0, 1)$
8:      **if** $u \leq \alpha^{(j)}$ **then**
9:         $\boldsymbol{\theta}^{(j)} = \boldsymbol{\theta}^*$
10:        $\hat{\boldsymbol{x}}_{1:T} = \hat{\boldsymbol{x}}_{1:T}^*$
11:     **else**
12:        $\boldsymbol{\theta}^{(j)} = \boldsymbol{\theta}^{(j-1)}$
13:     **end if**
14:      Compute $\eta^{(j)} = j^{-\gamma}$
15:      Compute $\mathbf{D}_j = \left(\mathbf{I} + \eta^{(j)}\left(\alpha^{(j)} - \bar{\alpha}_*\right)\frac{\mathbf{a}\mathbf{a}^\top}{||\mathbf{a}||^2}\right)$
16:      Compute $\mathbf{S}_j = \text{Chol}\left(\mathbf{S}_{j-1}\mathbf{D}_j\mathbf{S}_{j-1}^\top\right)$
17: **end for**
18: State filtering $\Rightarrow \hat{\boldsymbol{x}}_{1:T}$
19: Parameter estimation with $\left\{\boldsymbol{\theta}^{(j)}\right\}_{j=1}^{M}$ as in (38) $\Rightarrow \hat{\boldsymbol{\theta}}$

---

4 Computer simulation results

We simulated the data of a neuron following the Morris-Lecar model. Particularly, we generated data sampled at $f_s = 4$ kHz. Notice that typical sampling rates are on the order of kilohertz, therefore ensuring that we are operating in the regime where the Nyquist rate is well satisfied (that is, $f_s > 2 \cdot \text{BW}$, with BW the bandwidth of the recorded signal) [1, Chapter 3]. The model parameters were set to $C_m = 20$ $\mu\text{F/cm}^2$, $\phi = 0.04$, $V_1 = -1.2$ mV, $V_2 = 18$ mV, $V_3 = 2$ mV, and $V_4 = 30$ mV; the reverse potentials were $E_L = -60$ mV, $E_{Ca} = 120$ mV, and $E_K = -84$ mV; and the maximal conductances were $\bar{g}_{Ca} = 4.4$ mS/cm^2 and $\bar{g}_K = 8.0$ mS/cm^2. We considered a measurement noise with a standard deviation of 1 mV, which corresponds to an SNR of 32 dB. This value is considered a reasonable value in nowadays intracellular sensing devices. Model inaccuracies were generated as in Section 2.4.

Three sets of simulations are discussed. First, we validated the filtering method considering perfect knowledge of the model. In this case, the method in Algorithm 1 was used. Secondly, the model assumptions were relaxed in the sense that the parameters of the model were not known by the method. We analyzed the capabilities of the proposed method to infer both the time-evolving states of the system and some of the parameters defining the model. In this case, the method in Algorithm 3 was used. Finally, we validated the performance of the proposed methods in inferring the synaptic conductances. We tested both PF and PMCMC methods, that is, with and without full knowledge of the model, respectively.

4.1 Model parameters are known

In the simulations, we considered the model inaccuracies described in Section 2.4. To excite the neuron into spiking activity, a nominal applied current was injected with $I_o = 110$ $\mu\text{A/cm}^2$ and two values for σ_I were considered, namely 1 and 10% of I_o. The nominal conductance used in the model was $\bar{g}_L = 2$ mS/cm^2, whereas the underlying neuron had a zero-mean Gaussian error with standard deviation $\sigma_{\bar{g}_L}$. Two variance values were considered as well, 1 and 10% of $\bar{g}_L$. Finally, we considered $\sigma_n = 10^{-3}$ in the dynamics of the gating variable.

To give some intuition on the operation and performance of the PF method in Algorithm 1, we show the results for a single realization in Fig. 2. The results are for 500 particles and two different values of $\sigma_{y,k}^2$, corresponding to 0 and 32 dB, respectively. Even in very low SNR regimes, the method is able to operate and provide reliable filtering results.

In order to evaluate the efficiency of the proposed estimation method, we computed the Posterior Cramér-Rao Bound (PCRB) [52] in Appendix 2. We plot the PCRB as a benchmark for the root mean square error (RMSE) curves obtained by computer simulations, obtained after averaging 200 independent Monte-Carlo trials. For a generic time series w_k, the RMSE of an estimator $\hat{w}_k$ is defined as

$$\text{RMSE}(w_k) = \sqrt{\mathbb{E}\left\{(w_k - \hat{w}_k)^2\right\}}$$

$$\approx \sqrt{\frac{1}{M}\sum_{j=1}^{M}(w_k - \hat{w}_{j,k})^2}, \quad (39)$$

where $\hat{w}_{j,k}$ denotes the estimate of w_k at the jth realization and M the number of independent Monte-Carlo trials used to approximate the mathematical expectation.

Figures 3 and 4 show the time course of the RMSE using $N = \{500, 1000\}$ particles. We see that in both scenarios, our method efficiently attains the PCRB. We measure the efficiency ($\eta \geq 1$) of the method as the quotient between the RMSE and the PCRB, averaged over the
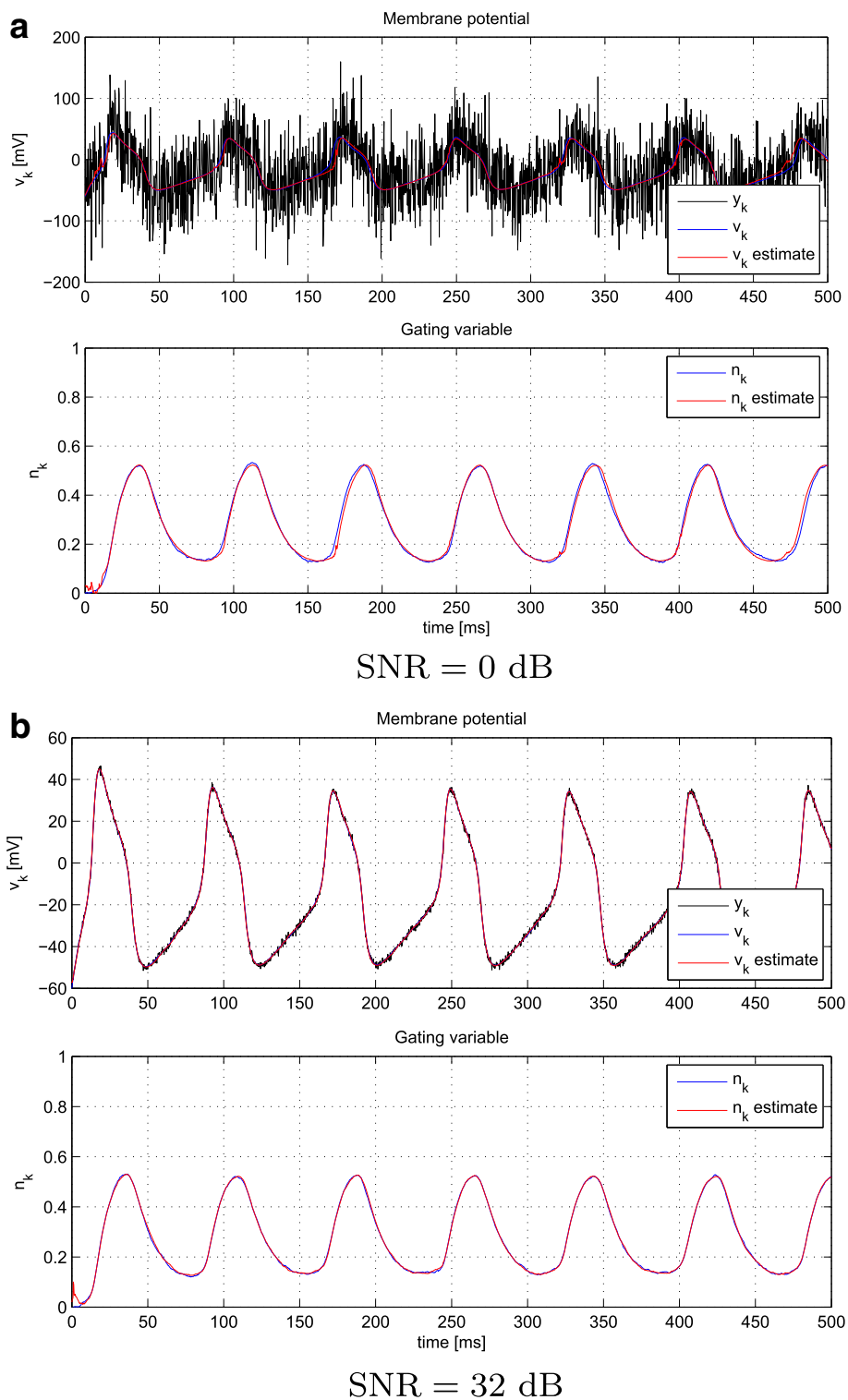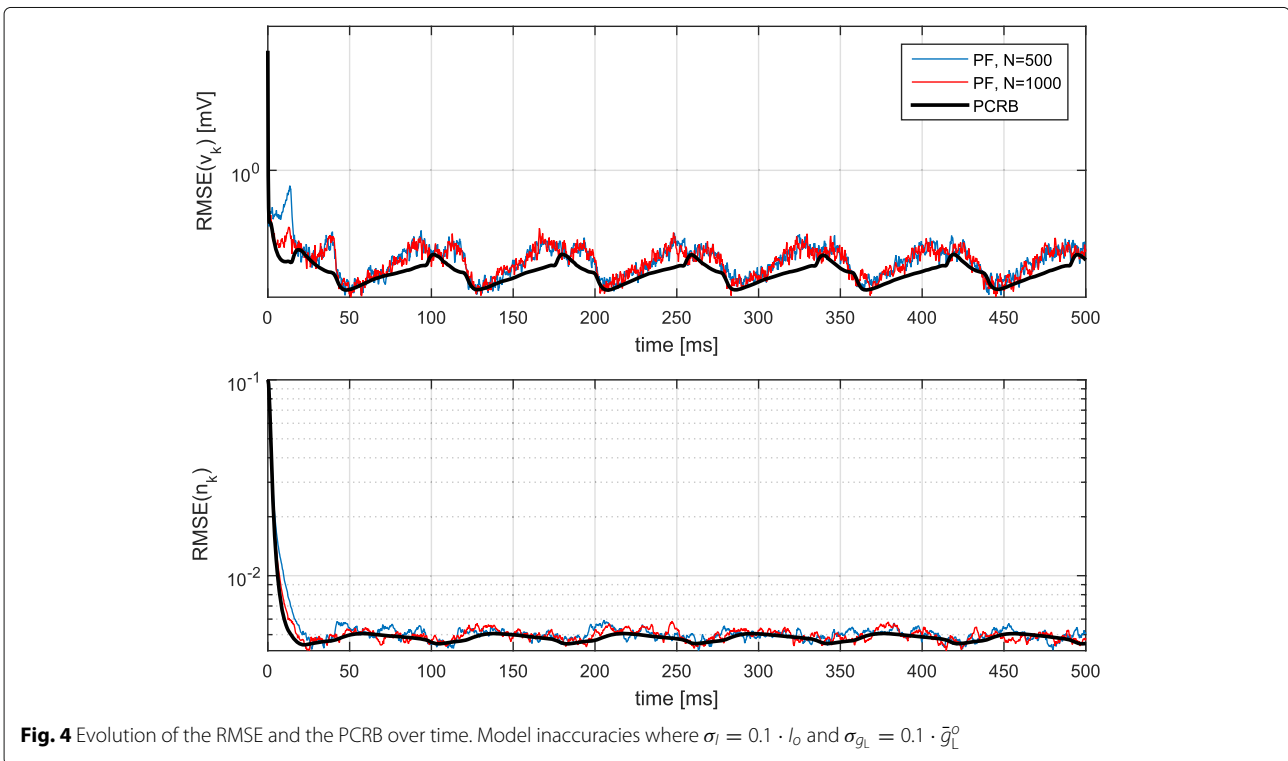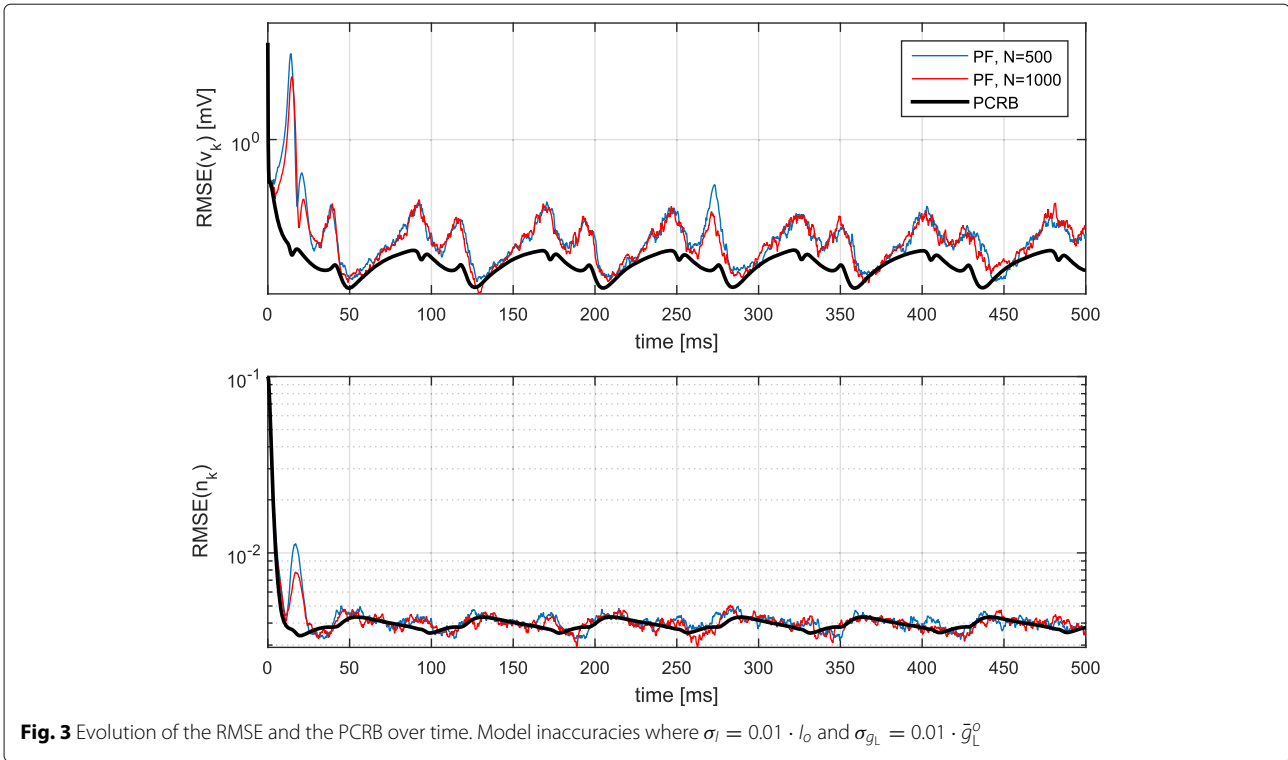
Fig. 2 A single realization of the PF method for **a** SNR = 0 dB and **b** SNR = 32 dB

entire simulation time. The worst efficiency on estimating v_k was 1.43 corresponding to 500 particles and 10% of inaccuracies (see Fig. 4), and the best was 1.11 for 1000 particles and 1% of errors (see Fig. 3). In estimating n_k, the discrepancy was even lower, 1.06 and 1.03 for maximum and minimum η, respectively. Notice that, for larger

Fig. 3 Evolution of the RMSE and the PCRB over time. Model inaccuracies where $\sigma_l = 0.01 \cdot l_o$ and $\sigma_{g_L} = 0.01 \cdot \bar{g}_L^o$

Fig. 4 Evolution of the RMSE and the PCRB over time. Model inaccuracies where $\sigma_l = 0.1 \cdot l_o$ and $\sigma_{g_L} = 0.1 \cdot \bar{g}_L^o$

inaccuracies, the method seems more reactive. This is because the covariance associated with the state-space has larger values, resulting in a more nervous filter and, consequently, with smaller convergence rates. As a conclusion, the PF tends to the PCRB with the number of particles. Also, the performance (both theoretical and empirical) could be improved if model inaccuracies are reduced, i.e., if the model parameters are better estimated at a previous stage. For the sake of completeness, we summarize the results in Table 1, where the average RMSE and PCRB along the 500-ms simulation are provided. It is apparent that increasing the number of particles from $N = 500$ to $N = 1000$ does not improve significantly the performance of the method.

4.2 Model parameters are unknown

In this section, we validate the algorithm presented in Section 3.2. According to the previous analysis, we deem that 500 particles are enough for the filter to provide reliable results. The parameters of the PMCMC algorithm were set to $\gamma = 0.9$ and $\bar{\alpha}_* = 0.234$.

Figure 5 shows the results for a single realization when a number of parameters in the nominal model are unknown. We considered one, two, and four unknown parameters. Each of the plots include $M = 100$ iterations of the MCMC showing the evolution of the parameter estimation (top) and the superimposed recorded voltage in black and the filtered voltage trace in red (bottom). Model inaccuracies are of 1%, similarly as in Fig. 3. In these plots, we omitted the results for the gating variable for the sake of clarity. The true and initial values used in the experiments, as well as the initial covariances assumed, are detailed in Table 2. From the plots, we can observe that the method performs reasonably well even in the case of estimating the model parameters at the same time it is filtering out the noise in the membrane voltage traces.

A biologically meaningful signal is the leakage current. In general, the leakage gathers those ionic channels that are not explicitly modeled and other non-modeled sources of activity. The parameters driving the leak current are $\bar{g}_L$ and E_L. We tested and validated the proposed PMCMC in an experiment where the leak parameters were estimated at the same time the filtering solution was computed. Moreover, the statistics of the process noise were estimated as well, $\Sigma_{x,k}$. In this case, we iterated the PMCMC method 1000 times and average the results over 100 Monte-Carlo independent trials. The results are shown in Fig. 6, where the RMSE performance of the PMCMC method is compared to the performance of the original PF with perfect knowledge of the model.

It can be observed that the filtering performances with perfect knowledge of the model and with estimation of parameters by PMCMC are similar. Moreover, both approaches attain the theoretical lower bound of accuracy given by the PCRB, which is derived from the true model; see Appendix 2.

In Fig. 7, validation results for the parameter estimation capabilities of the PMCMC are shown. Particularly, we plotted a number of independent realizations of the sample trajectories $\left\{ \boldsymbol{\theta}^{(j)} \right\}_{j=1}^{M}$. We observe that all of them converge to the true values of the parameter. Recall that these true values were $\boldsymbol{\theta} = (\bar{g}_L, E_L)^\top = (2, -60)^\top$. The average of these realizations is given in Fig. 7, where the aforementioned convergence to the true parameter is highlighted.

4.3 Estimation of synaptic conductances

Finally, once the methods to estimate state variables and unknown parameters were consolidated, we proceeded to test the methods to our ultimate goal: estimating jointly the intrinsic states of the neuron and the extrinsic inputs (i.e., the synaptic conductances).

First, the method with perfect knowledge of the model was validated in Fig. 8. It can be observed that the intrinsic signals can be effectively recovered as before where synaptic inputs were not accounted for. The estimation of $g_E(t)$ and $g_I(t)$ is quite accurate, and the presence of spikes does not degrade the estimation capabilities of the method.

The PMCMC algorithm was tested similarly. In this case, we assumed that the model parameters related to v_k and n_k were accurately estimated, for instance, using an off-line procedure or that analyzed in Section 4.2. Therefore, we focused on the estimation of those parameters that describe the OU process of each of the synaptic terms. Particularly, we considered the values in Table 3. The results can be consulted in Fig. 9 and compared to those in Fig. 8.

In Fig. 10, we compare the estimated excitatory and inhibitory synaptic conductances with the actual ones. We show both the performance of the PF method (round circles) and the PMCMC method (red dots). We appreciate a slightly better performance of the PF method, which

Table 1 Averaged results over simulation time

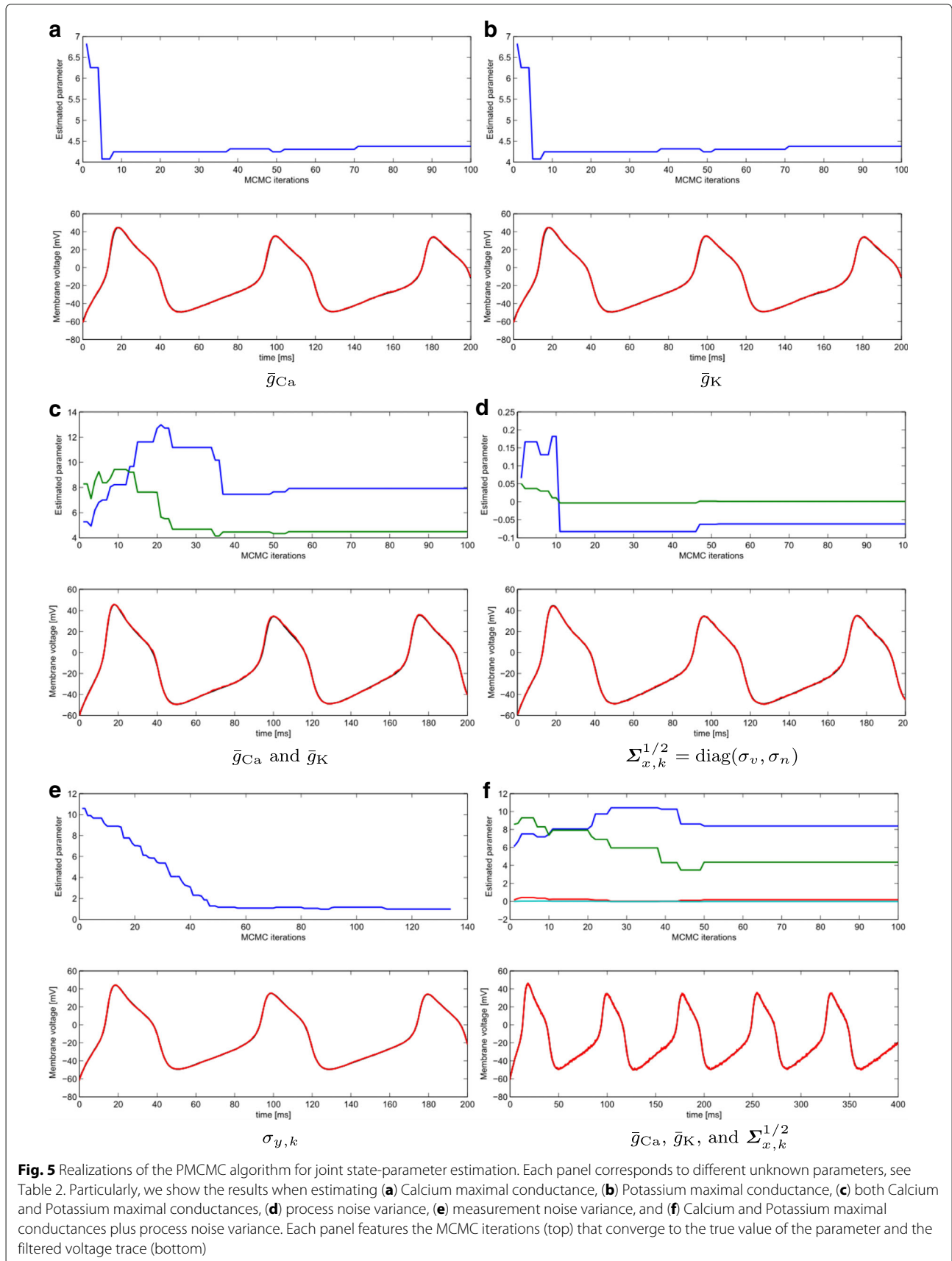| | $\sigma_I = 0.01 \cdot I_o,$ | | $\sigma_I = 0.1 \cdot I_o,$ | |
| | $\sigma_{g_L} = 0.01 \cdot \bar{g}_L^o$ | | $\sigma_{g_L} = 0.1 \cdot \bar{g}_L^o$ | |
	$N = 500$	$N = 1000$	$N = 500$	$N = 1000$
$\langle \text{RMSE}(v_k) \rangle$	0.3344	0.3211	0.4269	0.4203
$\langle \text{PCRB}(v_k) \rangle$	0.2325	0.2325	0.3777	0.3777
$\langle \text{RMSE}(n_k) \rangle$	0.0046	0.0045	0.0056	0.0055
$\langle \text{PCRB}(n_k) \rangle$	0.0043	0.0043	0.0053	0.0053

Fig. 5 Realizations of the PMCMC algorithm for joint state-parameter estimation. Each panel corresponds to different unknown parameters, see Table 2. Particularly, we show the results when estimating (**a**) Calcium maximal conductance, (**b**) Potassium maximal conductance, (**c**) both Calcium and Potassium maximal conductances, (**d**) process noise variance, (**e**) measurement noise variance, and (**f**) Calcium and Potassium maximal conductances plus process noise variance. Each panel features the MCMC iterations (top) that converge to the true value of the parameter and the filtered voltage trace (bottom)

Table 2 True value, initial value, and covariance of the parameters in Fig. 5

Parameter	True value	Initial value	Init. covariance
$\bar{g}_{Ca}$	4.4	8	1
$\bar{g}_{K}$	8	5	1
σ_v	0.0307	0.05	0.01
σ_n	0.001	0.01	0.001
$\sigma_{y,k}$	1	10	0.5

is normal since in this case a perfect knowledge of the model is assumed while the PMCMC needs to estimate both model parameters and synaptic conductances. More precisely, we have computed the normalized error

$$\frac{\sqrt{\sum_k \left(g_u(t_k) - \hat{g}_u(t_k) \right)^2}}{\sqrt{\sum_k g_u(t_k)^2}}$$

in both cases and obtained values of 0.2957 and 0.1939 for the excitatory and inhibitory conductances, respectively, with the PF method, and 0.3313 and 0.2095 (excitatory and inhibitory, respectively) with the PMCMC method. Compared to previous results in the literature, see, for instance, Figs. 2, 3, 4, 5, 6, and 8 in [4], where the authors performed an exhaustive comparison with different methods, our estimations provide excellent agreement with the prescribed conductances. Moreover, our statistics include data obtained in the spiking regime while the methods compared in [4] are applied only to subthreshold voltage traces. We also observe that the errors for the PMCMC method are, in average, only 12% (excitatory) and 8% (inhibitory) worse than the errors for the PF method. Given the complexity of the PMCMC method, these findings encourage future applications of the method to experimental data even when the model is not well defined.

We refer to the Additional file 1 to visualize a dynamic simulation showing how the estimations evolve as the PMCMC algorithm was applied in a case where the values of $\bar{g}_L$ and E_L were unknown.

5 Conclusions

In this paper, we propose a filtering method that is able to sequentially infer the time course of the membrane potential, the intrinsic activity of ionic channels, and the input synaptic conductances from noisy observations of voltage traces. The method works both for subthreshold and spiking regimes. It is based on the PF methodology

and features an optimal importance density, providing enhanced use of the particles that characterize the filtering distribution. In addition, we tackle the problem of joint parameter estimation and state filtering by extending the designed PF with an MCMC procedure in an iterative method known as PMCMC. Another distinctive contribution with respect to the other works in the literature is that here, we provide accuracy bounds for the problem at hand, given by the PCRB. The RMSEs of our methods are then compared to the bound, and therefore, we can assess the efficiency of the proposed inference methods.

Filtering methods of different types (e.g., PF or sigma-point Kalman filtering) have been used in other recent contributions to similar problems; see [7–13]. From a methodological perspective, the novelty of this paper is in the use of an optimal importance density to generate particles, a fact that increases the estimation accuracy for a given budget of particles. This technical detail only applies to PF methods. Although Gaussian methods (e.g., the family of sigma-point Kalman filters) have a lower computational cost in general, they require Gaussianity of the measures, whereas PFs do not. This is an advantage that we think can be crucial in estimating synaptic conductances, since the assumption of Gaussianity is generally assumed in the literature [4, 33], but there are no conclusive evidences to assert this assumption. In this paper, we have still applied the PF to an Ornstein-Uhlenbeck process in order to check that basic results can be attained, but further research will go in the direction of assuming other types of distributions for the synaptic conductances. The use of more complex distributions nicely fits within the framework of our PF-based method. Another advantage of PFs versus Gaussian filters is their enhanced robustness to outliers [53]; for instance, due to recording artifacts, future applications shall also incorporate this feature.

We have found excellent estimations of the synaptic conductances, even in spiking regimes. Estimating synaptic conductances in spiking regimes is a challenge which is far to be solved. It is well known that linear estimations of synaptic conductances are not trustable in this situation when data is extracted intracellularly from the spiking activity of neurons; see [14]. In experiments, thus, caution has to be taken in eliminating part of the voltage traces, thus losing also part of the temporal information of both excitatory and inhibitory conductances. Our method is able to perform well in this regime. This information is highly valuable in problems (epilepsy, schizophrenia, Alzheimer's disease, etc.) where a debate on the balance of excitation and inhibition is open; see the introduction of [12] for a rather complete overview on this feature.

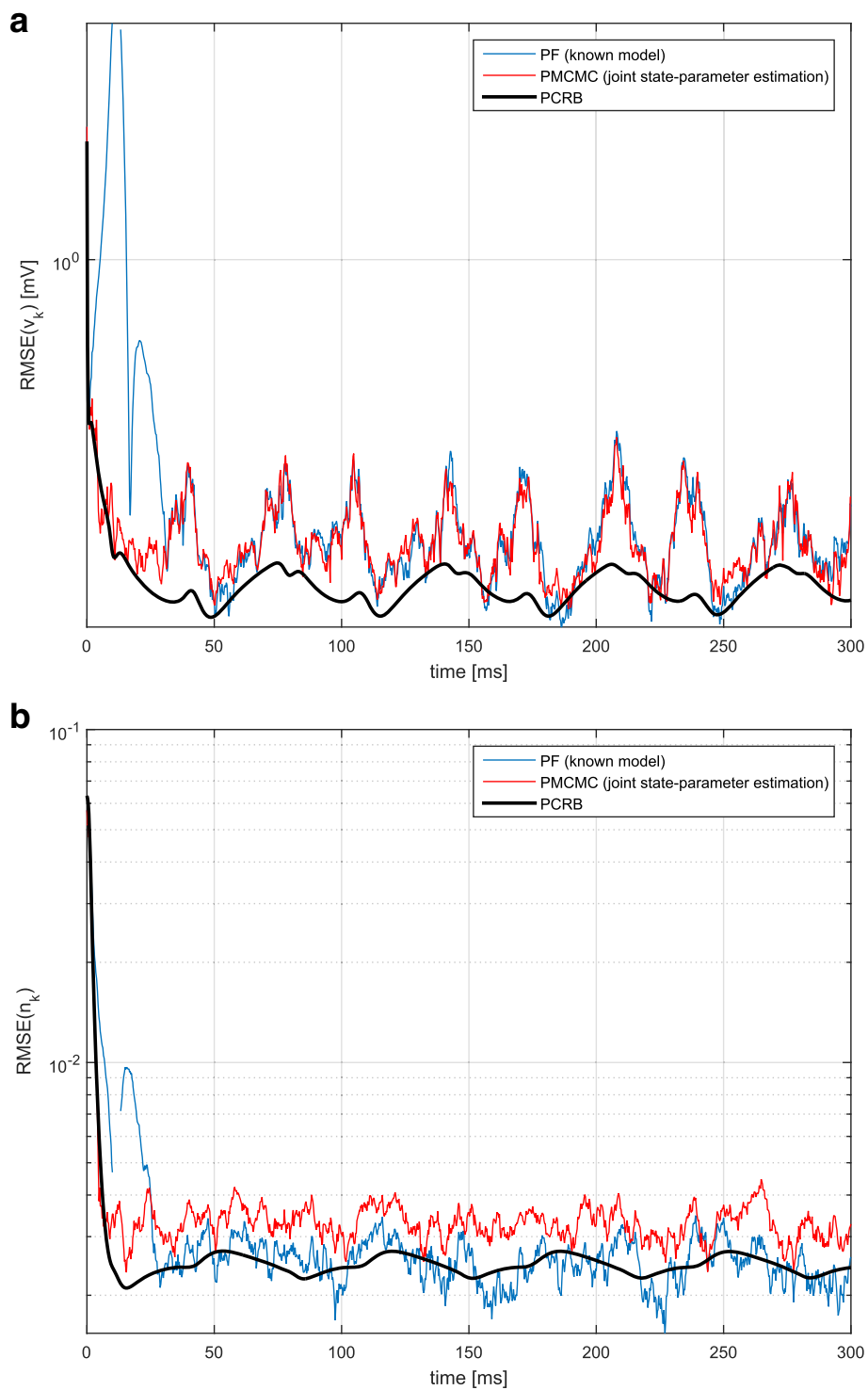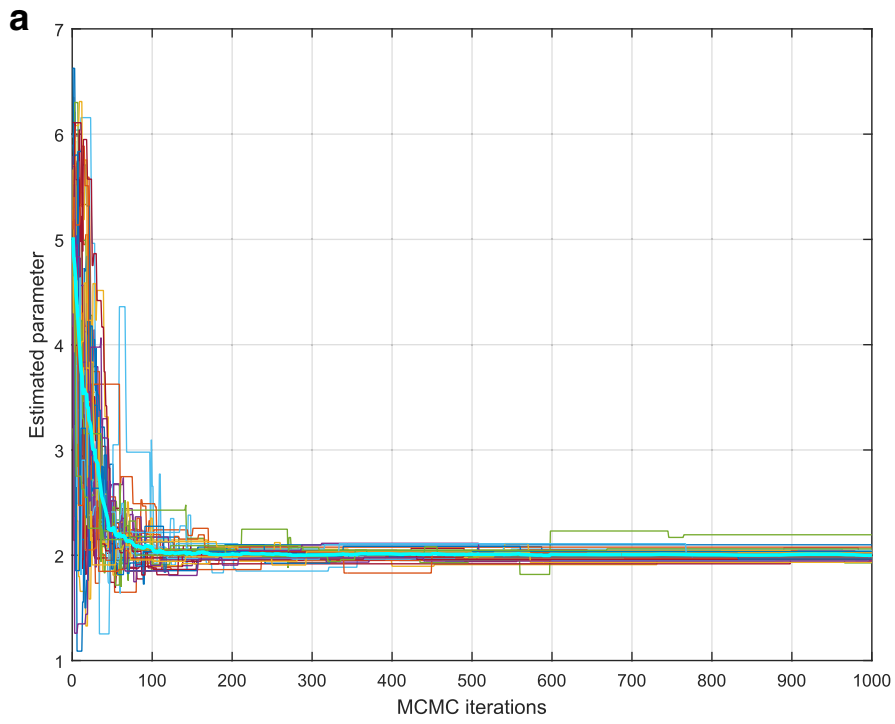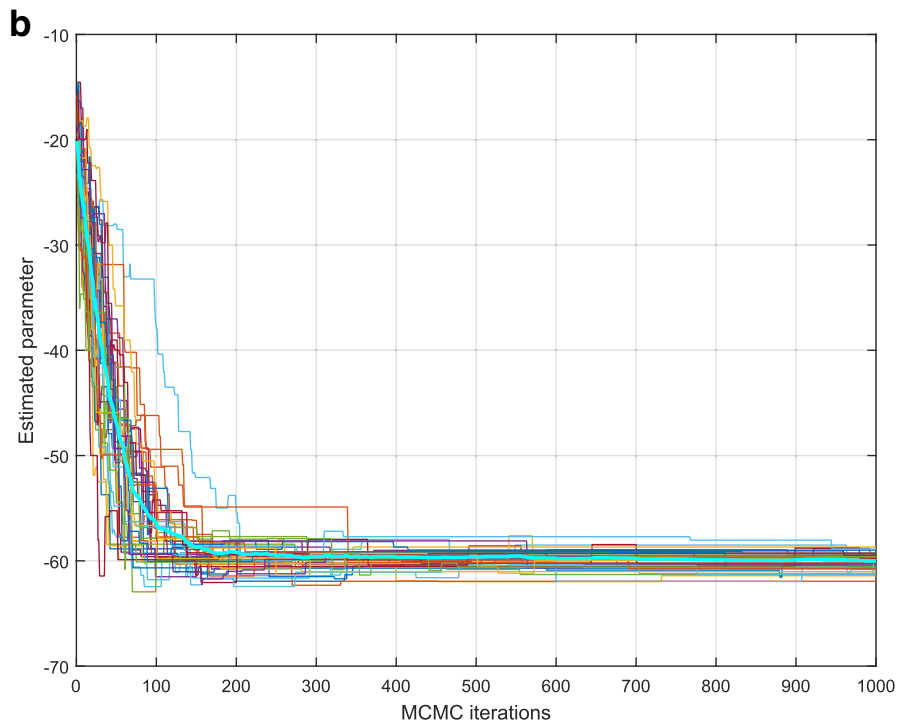The results show the validity of the approach when applied to a Morris-Lecar type of neuron. However, the

Fig. 6 Evolution of **a** RMSE(v_k) and **b** RMSE(n_k) over time for the PMCMC method estimating the leakage parameters. Model inaccuracies where $\sigma_l = 0.1 \cdot l_o$ and $\sigma_{g_L} = 0.1 \cdot \bar{g}_L^o$

Fig. 7 Parameter estimation performance of the proposed PMCMC algorithm. **a** Top plot shows results for $\bar{g}_L = 2$ estimation and (**b**) bottom plot for $E_L = -60$. Both plots show superimposed independent realizations together the average estimate of the parameter (thicker line)
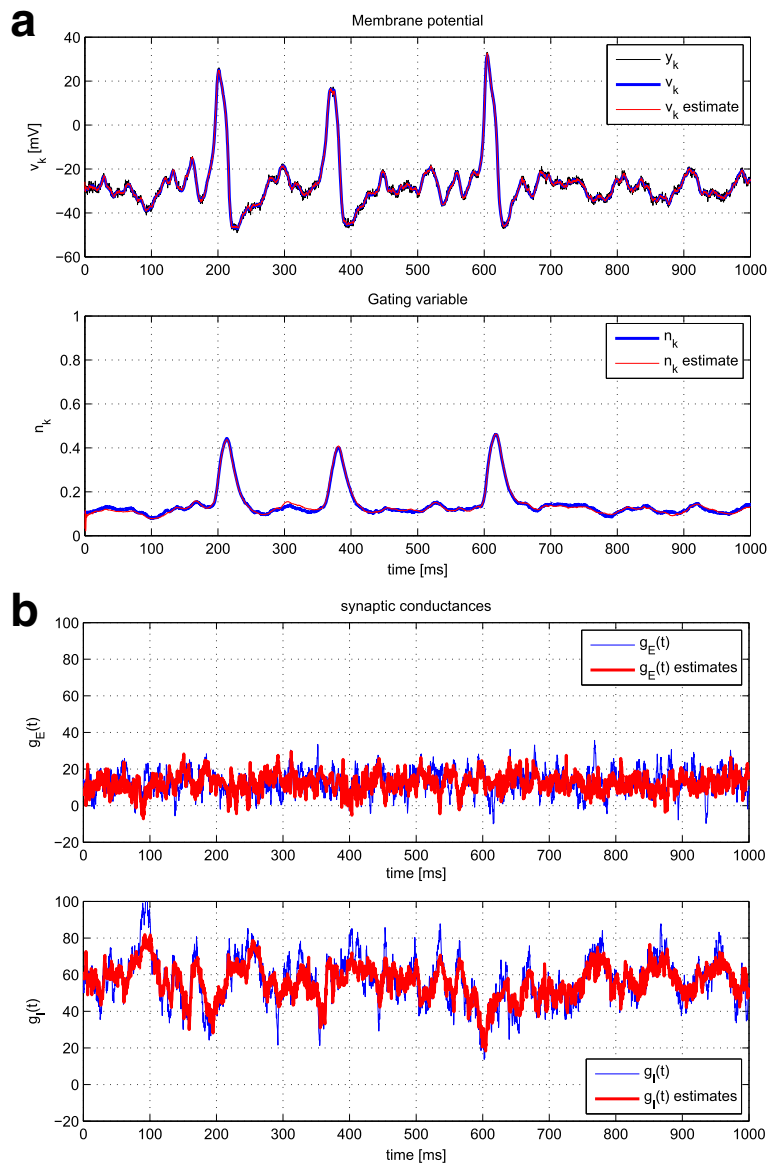
Fig. 8 A single realization of the PF method with perfect model knowledge, estimating voltage, and gating variables (**a**) and synaptic conductances in nS (**b**). See Table 3 for the true value, initial value, and covariance of the parameters used here

Table 3 True value, initial value, and covariance of the parameters in Fig. 9

Parameter	True value	Initial value	Init. covariance
τ_E	2.73	1.5	1
$g_{E,0}$	12.1	10	1
σ_E	12	25	5
τ_I	10.49	15	10
$g_{I,0}$	57.3	45	10
σ_I	26.4	35	5

procedure is general and could be applied to any neuron model exhibiting more complex dynamics like bursting and mixed-mode oscillations. Nevertheless, a clear drawback is the need for specifying a model although its parameters are estimated by the method. This is an ubiquitous problem in other model-based methods. Future research includes the enhancement of the proposed method to account for model variability, for instance, the use of Interacting Multiple Model (IMM) approaches. Other forthcoming applications could be validating the method using real data recordings, both for inferring
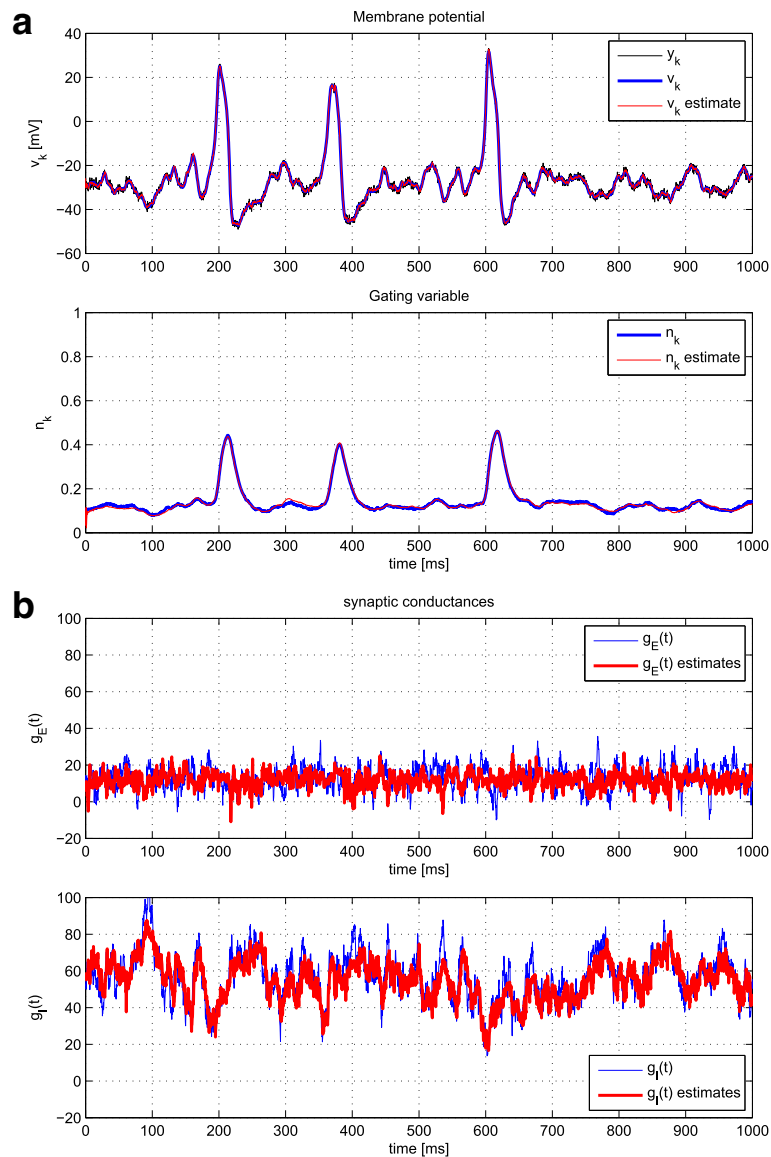
Fig. 9 A single realization of the PMCMC method, estimating voltage and gating variables (**a**) and synaptic conductances in nS (**b**) as well as model parameters. See Table 3 for the true value, initial value, and covariance of the parameters used here

parameters of the model and synaptic conductances. The latter problem is a challenging *hot* topic in the neuroscience literature, which is recently focusing on methods to extract the conductances from single-trace measurements. We think that our PF method would give useful and interesting results to physiologists that aim at inferring the brain's activation rules from neurons' activities. Actually, knowing the excitatory-inhibitory time-course separation can help in getting important conclusions about the brain's functional connectivity (see [54–56]).

We have not tried to obtain estimations when subthreshold ionic currents are active, where the presence of nonlinearities could also contaminate the estimations; see [15]. According to the excellent performance in spiking regimes, where nonlinearities are stronger, we expect also a good agreement between the estimated data and the prescribed synaptic conductances. Other extensions of the model can be devoted to incorporate the dendro-somatic interaction (see, for instance, [57]), by considering multi-compartment neuron models, thus taking into account the morphology and the functional properties of the cell. This is another big challenge for which we think that our method can account for.
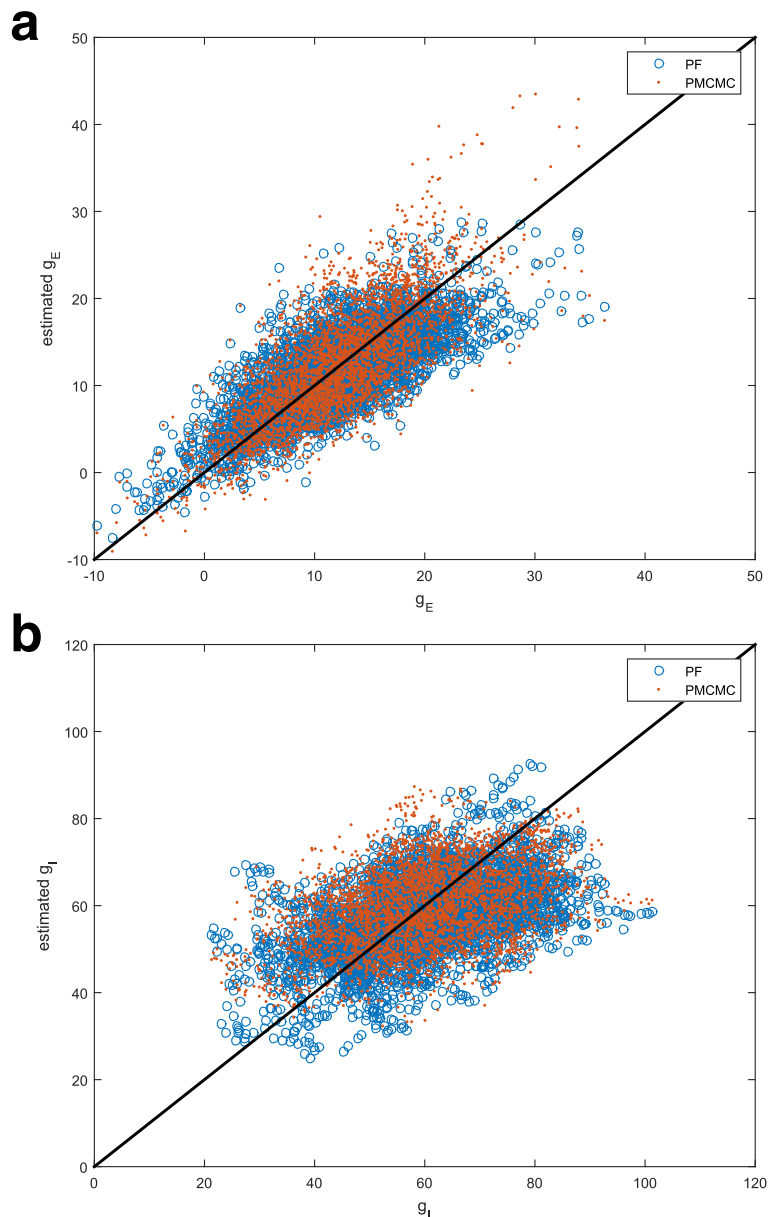
Fig. 10 Comparison of estimated excitatory (**a**) and inhibitory (**b**) synaptic conductances versus the actual ones. We show both the performance of the PF method (round circles), which assumes a perfect knowledge of the model, and the PMCMC method (red dots)

Appendix 1: Morris-Lecar neuron model

From the myriad of existing single-neuron models, we consider without loss of generality the Morris-Lecar model proposed in [34]. The model can be related (see [36]) to the $I_{Na,p} + I_K$ model (persistent sodium plus potassium model). The dynamics of the neuron is modeled by a continuous-time dynamical system composed of the current-balance equation for the membrane potential, $v = v(t)$, and the K$^+$ gating variable $0 \leq n = n(t) \leq 1$,

which represents the probability of the K$^+$ ionic channel to be active. Then, the system of differential equations is

$$C_m \dot{v} = -I_L - I_{Ca} - I_K + I_{app} \tag{40}$$

$$\dot{n} = \phi \frac{n_\infty(v) - n}{\tau_n(v)}, \tag{41}$$

where C_m is the membrane capacitance and ϕ a non-dimensional constant. I_{app} represents the (externally) applied current. For the time being, we have neglected I_{syn}

in (40). The leakage, calcium, and potassium currents are of the form

$$I_L = \bar{g}_L(v - E_L) \tag{42}$$

$$I_{Ca} = \bar{g}_{Ca} m_\infty(v)(v - E_{Ca}) \tag{43}$$

$$I_K = \bar{g}_K n(v - E_K), \tag{44}$$

respectively. $\bar{g}_L$, $\bar{g}_{Ca}$, and $\bar{g}_K$ are the maximal conductances of each current. E_L, E_{Ca}, and E_K denote the Nernst equilibrium potentials, for which the corresponding current is zero, a.k.a. reverse potentials.

The dynamics of the activation variable m is considered at the steady state, and thus, we write $m = m_\infty(v)$. On the other hand, the time constant $\tau_n(v)$ for the gating variable n cannot be considered that fast and the corresponding differential equation needs to be considered. The formulae for these functions are

$$m_\infty(v) = \frac{1}{2} \cdot \left(1 + \tanh\left[\frac{v - V_1}{V_2}\right]\right) \tag{45}$$

$$n_\infty(v) = \frac{1}{2} \cdot \left(1 + \tanh\left[\frac{v - V_3}{V_4}\right]\right) \tag{46}$$

$$\tau_n(v) = 1 / \left(\cosh\left[\frac{v - V_3}{2V_4}\right]\right), \tag{47}$$

which parameters V_1, V_2, V_3, and V_4 can be measured experimentally [36].

The knowledgeable reader would have noticed that the Morris-Lecar model is a Hodgin-Huxley-type model with the usual considerations, where the following two extra assumptions were made: the depolarizing current is generated by Ca^{2+} ionic channels (or Na^+ depending on the type of neuron modeled), whereas hyperpolarization is carried by K^+ ions, and that $m = m_\infty(v)$. The Morris-Lecar model is very popular in computational neuroscience as it models a large variety of neural dynamics while its phase-plane analysis is more manageable as it involves only two states [35].

The Morris-Lecar, although simple to formulate, results in a very interesting model as it can produce a number of different dynamics. For instance, for given values of its parameters, we encounter a subcritical Hopf bifurcation for $I_{app} = 93.86\ \mu A/cm^2$. On the other hand, for another set of parameter values, the system of equations has a Saddle-Node on an Invariant Circle (SNIC) bifurcation at $I_{app} = 39.96\ \mu A/cm^2$.

Appendix 2: PCRB in Morris-Lecar models

This appendix is devoted to the derivation of the PCRB estimation bound for the Morris-Lecar model used to benchmark the proposed methods in the simulations. We follow the sequential procedure given in [58], accounting that we have nonlinear functions in the state evolution and linear measurements, both with additive Gaussian noise. We are interested in an estimation error bound of the type of

$$\mathbb{E}_{y_k, x_k}\left\{\left(\hat{x}_k(y_{1:k}) - x_k\right)\left(\hat{x}_k(y_{1:k}) - x_k\right)^\top\right\} \geq J_k^{-1}, \tag{48}$$

where $\hat{x}_k(y_{1:k})$ represents an estimator of x_k given $y_{1:k}$.

Recall that the state-space we are dealing with is of the form

$$x_k = \mathbf{f}_{k-1}(x_{k-1}) + v_k$$
$$y_k = \mathbf{h}x_k + e_k, \tag{49}$$

where $\mathbf{h} = (1, 0)$, $x_k = (v_k, n_k)^\top$, and $\mathbf{f}_{k-1}(x_{k-1})$ defined by (8) and (9). The noise terms are of the form

$$v_k \sim \mathcal{N}(\mathbf{0}, \Sigma_{x,k}) \tag{50}$$

$$e_k \sim \mathcal{N}(0, \sigma_{y,k}^2). \tag{51}$$

In this case, the PCRB can be computed recursively by virtue of the result in [58] by computing the following terms

$$\mathbf{D}_k^{11} = \mathbb{E}_{x_k}\left\{\tilde{\mathbf{F}}_k^\top \Sigma_{x,k}^{-1} \tilde{\mathbf{F}}_k\right\} \tag{52}$$

$$\mathbf{D}_k^{12} = \mathbf{D}_k^{21} = -\mathbb{E}_{x_k}\left\{\tilde{\mathbf{F}}_k^\top\right\}\Sigma_{x,k}^{-1} \tag{53}$$

$$\mathbf{D}_k^{22} = \Sigma_{x,k}^{-1} + \mathbf{H}_{k+1}^\top \Sigma_{y,k+1}^{-1} \mathbf{H}_{k+1} \tag{54}$$

and plugging them into

$$\mathbf{J}_{k+1} = \mathbf{D}_k^{22} - \mathbf{D}_k^{21}\left(\mathbf{J}_k + \mathbf{D}_k^{11}\right)^{-1}\mathbf{D}_k^{12}, \tag{55}$$

for some initial $\mathbf{J}_0$. Notice that, in our case, $\mathbf{D}_k^{22}$ becomes deterministic, but the rest of the terms involving expectations should be computed by Monte-Carlo integration over independent state trajectories.

Since the state function is nonlinear, we use the Jacobian evaluated at the true value of x_k instead, that is,

$$\tilde{\mathbf{F}}_k = \left[\nabla_{x_k}\mathbf{f}_k^\top(x_k)\right]^\top = \begin{pmatrix} \frac{\partial f_1}{\partial v_t} & \frac{\partial f_1}{\partial n_t} \\ \frac{\partial f_2}{\partial v_t} & \frac{\partial f_2}{\partial n_t} \end{pmatrix}, \tag{56}$$

where functions f_1 and f_2 are as in (8) and (9), respectively. Therefore, to evaluate the bound, we need to compute the derivatives in the Jacobian. These are

$$\frac{\partial f_1(x_k)}{\partial v_k} =$$
$$1 - \frac{T_s}{C_m}\left(\bar{g}_L + \bar{g}_K n_k + \bar{g}_{Ca}\frac{\partial m_\infty(v_k)}{\partial v_k}v_k + \bar{g}_{Ca}m_\infty(v_k)\right)$$

$$\frac{\partial f_2(x_k)}{\partial v_k} =$$
$$T_s\phi\frac{\frac{\partial n_\infty(v_k)}{\partial v_k}\tau_n(v_k) - (n_\infty(v_{k-1}) - n_{k-1})\frac{\partial \tau_n(v_k)}{\partial v_k}}{\tau_n^2(v_k)}$$

$$\frac{\partial f_1(x_k)}{\partial n_k} = -\frac{T_s}{C_m}\bar{g}_K(v_k - E_K)$$

$$\frac{\partial f_2(x_k)}{\partial n_k} = 1 - \frac{T_s\phi}{\tau_n(v_k)}$$

with

$$\frac{\partial m_\infty(v_k)}{\partial v_k} = \frac{1}{2V_2}\text{sech}^2\left(\frac{v_k - V_1}{V_2}\right) \tag{57}$$

$$\frac{\partial n_\infty(v_k)}{\partial v_k} = \frac{1}{2V_4}\text{sech}^2\left(\frac{v_k - V_3}{V_4}\right) \tag{58}$$

$$\frac{\partial \tau_n(v_k)}{\partial v_k} = -\frac{1}{2V_4}\frac{\sinh\left(\frac{v_k - V_3}{2V_4}\right)}{\cosh^2\left(\frac{v_k - V_3}{2V_4}\right)}. \tag{59}$$

Additional file

Additional file 1: Evolution of the iterative learning when estimating leakage parameters. (MP4 2129 KB)

Authors' contributions
Both authors contributed equally to this work. Both authors read and approved the final manuscript.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details
[1]Northeastern University, Boston, MA 02115, USA. [2]Universitat Politècnica de Catalunya, 08028 Barcelona, Spain.

References
1. R Brette, A Destexhe, *Handbook of neural activity measurement.* (Cambridge University Press, New York, 2012). http://dx.doi.org/10.1017/CBO9780511979958
2. QJM Huys, L Paninski, Smoothing of and parameter estimation from noisy biophysical recordings. PLoS Comput. Biol. **5**(5), 1–16 (2009)
3. S Ditlevsen, A Samson, Estimation in the partially observed stochastic Morris-Lecar neuronal model with particle filter and stochastic approximation methods. Ann. Appl. Stat. **8**(2), 674–702 (2014). doi:10.1214/14-AOAS729
4. M Lankarany, W-P Zhu, MNS Swamy, Joint estimation of states and parameters of Hodgkin-Huxley neuronal model using Kalman filtering. Neurocomputing. **136**, 289–299 (2014). http://dx.doi.org/10.1016/j.neucom.2014.01.003
5. S Ditlevsen, A Samson, Parameter estimation in neuronal stochastic differential equation models from intracellular recordings of membrane potentials in single neurons: a review. J. de la Socié. **157**(1), 6–16 (2016)
6. Y Mishchenko, JT Vogelstein, L Paninski, A Bayesian approach for inferring neuronal connectivity from calcium fluorescent imaging data. Ann. Appl. Stat. **5**(2B), 1229–1261 (2011). doi:10.1214/09-AOAS303. http://dx.doi.org/10.1214/09-AOAS303
7. M Rudolph, Z Piwkowska, M Badoual, T Bal, A Destexhe, A method to estimate synaptic conductances from membrane potential fluctuations. J. Neurophysiol. **91**(6), 2884–2896 (2004). doi:10.1152/jn.01223.2003. jn.physiology.org/content/91/6/2884.full
8. M Pospischil, M Toledo-Rodriguez, C Monier, Z Piwkowska, Thierry Bal, Y Frégnac, H Markram, A Destexhe, Minimal Hodgkin-Huxley type models

for different classes of cortical and thalamic neurons. Biol. Cybernet. **99**(4-5), 427–441 (2008)
9. C Bédard, S Béhuret, C Deleuze, T Bal, A Destexhe, Oversampling method to extract excitatory and inhibitory conductances from single-trial membrane potential recordings. J. Neurosci. Methods (2011). doi:10.1016/j.jneumeth.2011.09.010
10. R Kobayashi, Y Tsubo, P Lansky, S Shinomoto, Estimating time-varying input signals and ion channel states from a single voltage trace of a neuron. Adv. Neural Inf. Process. Syst. (NIPS). **24**, 217–225 (2011)
11. L Paninski, M Vidne, B DePasquale, DG Ferreira, Inferring synaptic inputs given a noisy voltage trace via sequential Monte Carlo methods. J. Comput. Neurosci. **33**(1), 1–19 (2012)
12. RW Berg, S Ditlevsen, Synaptic inhibition and excitation estimated via the time constant of membrane potential fluctuations. J. Neurophysiol. **110**(4), 1021–1034 (2013). doi:10.1152/jn.00006.2013
13. M Lankarany, WP Zhu, MNS Swamy, T Toyoizumi, Inferring trial-to-trial excitatory and inhibitory synaptic inputs from membrane potential using Gaussian mixture Kalman filtering. Front. Comput. Neurosci. **7** (2013). doi:10.3389/fncom.2013.00109. http://dx.doi.org/10.3389/fncom.2013.00109
14. A Guillamon, DW McLaughlin, J Rinzel, Estimation of synaptic conductances. J. Physiology-Paris. **100**(1-3), 31–42 (2006). doi:10.1016/j.jphysparis.2006.09.010
15. C Vich, A Guillamon, Dissecting estimation of conductances in subthreshold regimes. J. Comput. Neurosci, 1–17 (2015). doi:10.1007/s10827-015-0576-2
16. P Closas, A Guillamon, in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2013.* Sequential estimation of gating variables from voltage traces in single-neuron models by particle filtering, (Vancouver, 2013)
17. P Closas, A Guillamon, Estimation of neural voltage traces and associated variables in uncertain models. BMC Neurosci. **14**(1), 1151 (2013)
18. A Doucet, N de Freitas, N Gordon, *Sequential Monte Carlo methods in practice.* (Springer, 2001)
19. PM Djuric, SJ Goodsill, Guest editorial special issue on Monte Carlo methods for statistical signal processing. Signal Process. IEEE Trans. **50**(2), 173–173 (2002). doi:10.1109/TSP.2002.978373
20. PM Djurić, JH Kotecha, J Zhang, Y Huang, T Ghirmai, MF Bugallo, J Míguez, Particle filtering. IEEE Signal Process. Mag. **20**(5), 19–38 (2003)
21. S Arulampalam, S Maskell, N Gordon, T Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Trans. Signal Process. **50**(2), 174–188 (2002)
22. Z Chen, Bayesian filtering: from Kalman filters to particle filters, and beyond. Technical report, Adaptive Syst. Lab., McMaster University, Ontario, Canada (2003)
23. B Ristic, S Arulampalam, N Gordon, *Beyond the Kalman filter: particle filters for tracking applications.* (Artech House, Boston, 2004)
24. S Särkkä, *Bayesian filtering and smoothing.* (Cambridge University Press, New York, 2013)
25. A Doucet, SJ Godsill, C Andrieu, On sequential Monte Carlo sampling methods for Bayesian filtering. Stat. Comput. **3**, 197–208 (2000)
26. DV Vavoulis, VA Straub, JAD Aston, J Feng, A self-organizing state-space-model approach for parameter estimation in Hodgkin-Huxley-type models of single neurons. PLoS Comput. Biol. **8**(3) (2012). e1002401
27. G Ullah, SJ Schiff, Tracking and control of neuronal Hodgkin-Huxley dynamics. Phys. Rev. E. **79**(4), 040901 (2009)
28. CM Carvalho, MS Johannes, HF Lopes, NG Polson, et al., Particle learning and smoothing. Stat. Sci. **25**(1), 88–106 (2010)
29. N Chopin, PE Jacob, O Papaspiliopoulos, SMC2: an efficient algorithm for sequential analysis of state space models. J. R. Stat. Soc. Series B (Stat. Methodol.) **75**(3), 397–426 (2013)
30. CC Drovandi, JM McGree, AN Pettitt, A sequential Monte Carlo algorithm to incorporate model uncertainty in Bayesian sequential design. J. Comput. Graphical Stat. **23**(1), 3–24 (2014)
31. I Urteaga, MF Bugallo, PM Djurić, in *Statistical Signal Processing Workshop (SSP) 2016 IEEE.* Sequential Monte Carlo methods under model uncertainty (IEEE, Mallorca, 2016), pp. 1–5
32. L Martino, J Read, V Elvira, F Louzada, Cooperative parallel particle filters for online model selection and applications to urban mobility. Digital Signal Process. **60**, 172–185 (2017)

33. M Rudolph, A Destexhe, Characterization of subthreshold voltage fluctuations in neuronal membranes. Neural Comput. **15**(11), 2577–2618 (2003)
34. C Morris, H Lecar, Voltage oscillations in the barnacle giant muscle fiber. Biophys J. **35**(1), 193–213 (1981)
35. JR Rinzel, GB Ermentrout, in *Methods in neural modeling*, ed. by C Koch, I Segev. Analysis of neural excitability and oscillations (MIT Press, Cambridge, 1998), pp. 135–169
36. E Izhikevich, *Dynamical systems in neuroscience: the geometry of excitability and bursting*. (MIT Press, Cambridge, 2006)
37. R Douc, O Cappé, E Moulines, in *Proc. of the 4th International Symposium on Image and Signal Processing and Analysis, ISPA'05*. Comparison of resampling schemes for particle filtering, (Zagreb, 2005), pp. 64–69
38. C Andrieu, A Doucet, SS Singh, VB Tadic, Particle methods for change detection, system identification, and control. Proc. IEEE. **92**(3), 423–438 (2004)
39. C Andrieu, A Doucet, VB Tadic, in *Decision and Control 2005 and 2005 European Control Conference.CDC-ECC '05. 44th IEEE Conference on*. On-line parameter estimation in general state-space models, (Seville, 2005), pp. 332–337
40. G Poyiadjis, A Doucet, SS Singh, Particle approximations of the score and observed information matrix in state space models with application to parameter estimation. Biometrika. **98**, 65–80 (2011)
41. C Andrieu, A Doucet, R Holenstein, Particle Markov Chain Monte Carlo methods. J. R. Stat. Soc. Series B. **72**(3), 269–342 (2010)
42. L Martino, V Elvira, G Camps-Valls, Group importance sampling for particle filtering and MCMC (2017). arXiv preprint arXiv:1704.02771
43. WR Gilks, S Richardson, DJ Spiegelhalter, *Markov Chain Monte Carlo in practice: interdisciplinary statistics. CRC Interdisciplinary Statistics Series*. (Chapman & Hall, 1996)
44. C Berzuini, N Best, W Gilks, C Larizza, Dynamic conditional independence models and Markov Chain Monte Carlo methods. J. Am. Stat. Assoc. **92**, 1403–1412 (1997)
45. JS Liu. Monte Carlo strategies in scientific computing (Springer, New York, 2008)
46. S Brooks, A Gelman, G Jones, X-L Meng, *Handbook of Markov Chain Monte Carlo*. (CRC Press, Boca Raton, 2011)
47. S Donnet, A Samson, Using PMCMC in EM algorithm for stochastic mixed models: theoretical and practical issues. J. Soc. Franç, aise de Statistique. **155**(1), 49–72 (2014)
48. M Vihola, Robust adaptive metropolis algorithm with coerced acceptance rate. Stat. Comput. **22**(5), 997–1008 (2012)
49. H Haario, E Saksman, J Tamminen, et al., An adaptive metropolis algorithm. Bernoulli. **7**(2), 223–242 (2001)
50. D Luengo, L Martino, in *Acoustics, Speech and Signal Processing (ICASSP) 2013 IEEE International Conference on*. Fully adaptive Gaussian mixture Metropolis-Hastings algorithm (IEEE, 2013), pp. 6148–6152
51. GH Golub, CF van Loan, *Matrix computations*, 3edition. (The John Hopkins University Press, Baltimore, 1996)
52. HL Van Trees, KL Bell, *Bayesian bounds for parameter estimation and nonlinear filtering/tracking*. (Wiley Interscience, Piscataway, 2007)
53. PM Djurić, MF Bugallo, P Closas, J Míguez, in *Proceedings of the IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, CAMSAP'09*. Measuring the robustness of sequential methods (Dutch Antilles, 2009)
54. JS Anderson, M Carandini, D Ferster, Orientation tuning of input conductance, excitation, and inhibition in cat primary visual cortex. J. Neurophysiol. **84**(2), 909–926 (2000). jn.physiology.org/content/84/2/909.short
55. M Wehr, AM Zador, Balanced inhibition underlies tuning and sharpens spike timing in auditory cortex. Nature. **426**(6965), 442–446 (2003). doi:10.1038/nature02116. http://dx.doi.org/10.1038/nature02116
56. C Bennett, S Arroyo, S Hestrin, Subthreshold mechanisms underlying state-dependent modulation of visual responses. Neuron. **80**(2), 350–357 (2013). http://dx.doi.org/10.1016/j.neuron.2013.08.007
57. SJ Cox, Estimating the location and time course of synaptic input from multi-site potential recordings. J. Comput. Neurosci. **17**, 225–243 (2004)
58. P Tichavský, CH Muravchik, A Nehorai, Posterior Cramér-Rao bounds for discrete-time nonlinear filtering. IEEE Trans. Signal Process. **46**(5), 1386–1396 (1998)