Biol Cybern (2012) 106:155–167
DOI 10.1007/s00422-012-0487-5

ORIGINAL PAPER

Biological
Cybernetics

Dynamical estimation of neuron and network properties II: path integral Monte Carlo methods

Mark Kostuk · Bryan A. Toth · C. Daniel Meliza ·
Daniel Margoliash · Henry D. I. Abarbanel

Received: 16 January 2012 / Accepted: 14 March 2012 / Published online: 13 April 2012

Abstract Hodgkin–Huxley (HH) models of neuronal membrane dynamics consist of a set of nonlinear differential equations that describe the time-varying conductance of various ion channels. Using observations of voltage alone we show how to estimate the unknown parameters and unobserved state variables of an HH model in the expected circumstance that the measurements are noisy, the model has errors, and the state of the neuron is not known when observations commence. The joint probability distribution of the observed membrane voltage and the unobserved state variables and parameters of these models is a path integral through the model state space. The solution to this integral allows estimation of the parameters and thus a characterization of many biological properties of interest, including channel complement and density, that give rise to a neuron's electrophysiological behavior. This paper describes a method for directly evaluating the path integral using a Monte Carlo numerical approach. This provides estimates not only of the expected values of model parameters but also of their posterior uncertainty. Using test data simulated from neuronal models comprising several common channels, we show that short (<50 ms) intracellular recordings from neurons stimulated with a complex time-varying current yield accurate and precise estimates of the model parameters as well as accurate predictions of the future behavior of the neuron. We also show that this method is robust to errors in model specification, supporting model development for biological preparations in which the channel expression and other biophysical properties of the neurons are not fully known.

Keywords Data assimilation · Neuronal dynamics · Ion channel properties · Markov Chain Monte Carlo

1 Introduction

The dynamical responses of neurons and nervous system networks to time-varying inputs depend on ion channels that are gated by membrane voltage and chemical ligands. The complex, nonlinear dynamics of the voltage-gated channels (Johnston and Wu 1995; Graham 2002) control the generation of action potentials and determine their shape and frequency. Determining which channels are present in a class of neuron, their biophysical properties, and how these contribute to the phenomenological behavior of the neurons is generally a painstaking process involving extensive pharmacological manipulation.

An alternative approach that we explore here is to record a neuron's response to a time-varying current and use these data to estimate properties of a biophysical model of the neuron. Accurate predictions of the neuron's response to novel stimuli by such a model indicates that the model may be used to make inferences about the biological properties of the neuron. The measurements are necessarily limited to a small subset of the many state variables in such a model, and there can be many unknown parameters. We have developed a systematic approach to this problem that provides an exact

M. Kostuk · B. A. Toth
Department of Physics, University of California, 9500 Gilman Drive, San Diego, La Jolla, CA 92093-0402, USA

C. D. Meliza (✉) · D. Margoliash
Department of Organismal Biology and Anatomy, University of Chicago, 1027 E 57th Street, Chicago, IL 60637, USA
e-mail: dmeliza@uchicago.edu

H. D. I. Abarbanel
Marine Physical Laboratory, Department of Physics (Scripps Institution of Oceanography), Center for Theoretical Biological Physics, University of California, 9500 Gilman Drive,
San Diego, La Jolla, CA 92093-0374, USA
e-mail: habarbanel@ucsd.edu

statistical setting for transferring information from measurements to a model, which we described in Part I of this paper (Toth et al. 2011), along with a variational approximation to the high-dimensional path integral involved in this approach.

Here, in Part II we move beyond the variational approximation, to a numerical evaluation (also approximate) of the integral over paths of the model state through and beyond the temporal window of observations. By sampling from the distribution of likely paths we can obtain an estimate of the state and parameter values that best represent the data, as well as statistics about the uncertainty of these estimates. This posterior uncertainty indicates how much the data, as well as the dynamics of the model, constrain the parameter and state estimates, thereby providing additional information that can be used to guide model selection and inference as well as experimental design.

Small error bounds indicate that the amount of data is sufficient to make confident statements about the underlying biophysical properties of the neuron; large error bounds for a parameter indicate either that the data is insufficient in quantity or dynamical range, that the behavior of the neuron is insensitive to the value of that parameter, or that the model is in error. We can also integrate forward in time beyond the observation interval using samples from the full posterior distribution to obtain a predictive distribution, which is useful for model validation and selection.

After a brief summary of the family of Hodgkin–Huxley (HH) biophysical models we utilize and of the path integral formulation of the data assimilation problem (for more detail see Toth et al. 2011), we describe a Markov Chain Monte Carlo (MCMC) algorithm for sampling from the distribution of paths conditioned on the recorded data. We also present an implementation of the algorithm suitable for highly parallelized devices such as graphical processing units (GPUs). We then repeat a number of the "twin experiments" from Part I, in which data simulated from a model is used in our procedure and the estimates are compared to the parameters used to generate the data as well as the observed and unobserved states.

The goal of these numerical experiments is to determine under what conditions the data assimilation procedure is accurate. We do not pursue two questions addressed in Part I related to the frequency of the input current and the amount of noise in the measurements, taking values that we demonstrated work well with the variational method. Instead we focus on the robustness of the method to model errors—when the model contains channels that are absent in the data and vice versa. As in Toth et al. (2011) we analyze two simple HH models: one has Na, K, and leak currents (NaKL model), and the other has in addition an I_h current (NaKLh model). We test for model robustness by generating data using one model and estimating parameters using the other, and show that we can identify missing or extra channels in the model. The numerical approach to evaluating the path integral described here is particularly suited to these situations, because it takes into account such model errors, which are inevitable in the study of biological neurons.

2 Methods

2.1 General framework

In studying the biological properties of neurons, we can typically measure only the membrane voltage potential, $V(t)$, while injecting a known stimulus current $I_{\mathrm{app}}(t)$. We would like to infer properties of the voltage-gated ion channels that open and close in response to the current injection. The task is to select a model that is consistent with the observations and to estimate the values of the parameters that have some connection to biologically interesting properties of the neuron and the system of which it is a part. Other types of measurements are possible, including voltage-clamp measurements of current flow or optical measurements from fluorescent reporters; these are not within the scope of the current paper. The basic problem, of estimating unobservable parameters and states from a limited set of observations, in principle remains the same, although the kinetics of some measurements (such as calcium indicators) may be sufficiently slow so as to challenge the modeling effort.

In making such inferences, models that are explicitly based on biophysical entities are preferable to more abstract phenomenological ones (e.g., integrate-and-fire). The biophysical models typically comprise a set of ordinary differential equations, for current conservation across the membrane (and possibly between different compartments) and for the kinetics of the gating variables that describe the opened and closed configurations of various voltage-gated channels (Sect. 2.5). In general, the system is described by a D-dimensional state vector that includes the observable voltage $V(t)$ and a set of unobservable variables $a_i(t)$; $i = 1, 2, \ldots, D-1$ associated with other compartments and the permeability of each of the voltage-gated channels included in the model. The models we examine in this paper are relatively simple, with a single compartment and up to five state variables associated with three channel types. The methods developed here and in our earlier paper (Toth et al. 2011) can be applied to specific, more complex settings, but will require richer models and possibly additional data.

The second component of the model is the process of observation. We make measurements of the voltage at discrete times over some interval t_n=$\{t_0, t_1, \ldots, t_m = T\}$. We label these observations $y(t_n) = y(n)$; $n = 0, 1, \ldots, m$, and they are related to the true voltage of the neuron through some measurement function $h(\mathbf{x})$ that incorporates noise arising from various sources. The discrete time nature of the observations suggests that the model be stated as a rule in discrete time taking the system at time $t_n = n$ to the state at time $n + 1$. This rule may be the specific formulation of a numerical solution algorithm for the underlying differential equations. Both the discrete time rule and the observation function may involve a collection of unknown parameters, which we denote as $\mathbf{p}$.

Using the observations $y(n)$ of the voltage, we wish to establish the full collection of state variables $\mathbf{x}(n)$ at times in the observation window, in particular at the end of the measurements $t = T$. With estimates of $\mathbf{x}(T)$, of the parameters $\mathbf{p}$, and knowledge of the stimulus for $t > T$ we may use the model to predict the neuron's behavior for $t > T$. Our goal is to achieve $y(t) \approx h(\mathbf{x}(t))$ for $t > T$, as these are the observed quantities. If these predictions are accurate for a broad range of biologically plausible stimuli, then the estimates of the model parameters provide a parsimonious, biophysically interpretable description of the neuron's behavior.

In the present paper the data are simulated from a model which has the form of a set of ordinary differential equations (Sect. 2.5), which we solve by discretizing time. After choosing a realistic set of parameters and initial states, we generate a solution to the equations. We select a subset of the output, here the voltage alone, and add noise to yield the observed quantities $y(n)$. This transfer of information from measurements to models is called *data assimilation*, following the name given in the extensive and well developed geophysical literature on this subject (Evensen 2009). The technique for testing data assimilation methods in cases where the data is generated by the model is known as a "twin experiment."

2.2 Path integral formulation

As discussed in Toth et al. (2011) one can cast the general set of questions in data assimilation when one has noisy data, model errors, and uncertainty about the initial state of the model $\mathbf{x}(t_0) = \mathbf{x}(0)$ into an integral over the states $\mathbf{x}(t_n) = \mathbf{x}(n)$; $\{t_0, t_1, \ldots, t_m = T\}$ during the measurement interval $[0, T]$ at the measurement times t_n. If we denote the path in state space through this time interval as $\mathbf{X} = \{\mathbf{x}(0), \mathbf{x}(1), \ldots, \mathbf{x}(m)\}$, then the path is a location in $(m + 1)D$-dimensional space.

The expected value, conditional on the measurements $\mathbf{Y} = \{\mathbf{y}(0), \mathbf{y}(1), \ldots, \mathbf{y}(m)\}$, of any function along the path $G(\mathbf{X})$ is given by the $(m + 1)D$ dimensional integral

$$E[G(\mathbf{X})|\mathbf{Y}] = \frac{\int d\mathbf{X} e^{-A_0(\mathbf{X},\mathbf{Y})} G(\mathbf{X})}{\int d\mathbf{X} e^{-A_0(\mathbf{X},\mathbf{Y})}}. \qquad (1)$$

The action $A_0(X, Y)$ is given in terms of (1) the conditional mutual information of a state $\mathbf{x}(n)$ and a measurement $\mathbf{y}(n)$, conditioned on measurements up to the time t_{n-1}: $\mathbf{Y}(n-1) = \{\mathbf{y}(0), \mathbf{y}(1), \ldots, \mathbf{y}(n-1)\}$, (2) the transition probability $P(\mathbf{x}(n + 1)|\mathbf{x}(n))$ to arrive in the state $\mathbf{x}(n + 1)$ at time t_{n+1} given the state $\mathbf{x}(n)$ at t_n, and (3) the distribution of states at t_0 $P(\mathbf{x}(0))$, as

$$\begin{aligned}
-A_0(\mathbf{X},\ \mathbf{Y}) &= \sum_{n=0}^{m} \log\left\{\frac{P(\mathbf{x}(n),\ \mathbf{y}(n)|\mathbf{Y}(n-1))}{P(\mathbf{x}(n)|\mathbf{Y}(n-1))\,P(\mathbf{y}(n)|\mathbf{Y}(n-1))}\right\} \\
&\quad + \sum_{n=0}^{m-1} \log[P(\mathbf{x}(n+1)|\mathbf{x}(n))] + \log[P(\mathbf{x}(0)] \\
&= \sum_{n=0}^{m} \log\left\{\frac{P(\mathbf{y}(n)|\mathbf{x}(n),\ \mathbf{Y}(n-1))}{P(\mathbf{y}(n)|\mathbf{Y}(n-1))}\right\} \\
&\quad + \sum_{n=0}^{m-1} \log[P(\mathbf{x}(n+1)|\mathbf{x}(n))] + \log[P(\mathbf{x}(0)]. \qquad (2)
\end{aligned}$$

The first sum gives the information transferred from the measurement $\mathbf{y}(n)$ to the state $\mathbf{x}(n)$, conditioned on previous measurements $\mathbf{Y}(n-1)$. The second term represents the underlying dynamics of the model that moves the state forward one step in time, and the last term is the uncertainty in the state at the time t_0 of the initial measurement.

Approximations to $A_0(\mathbf{X},\ \mathbf{Y})$ were discussed in Toth et al. (2011):

- If the dynamical rule is written as $g_a(\mathbf{x}(n + 1),\ \mathbf{x}(n),\ \mathbf{p}) = 0$; $a = 1, 2, \ldots, D$ in the case of no model errors, then the transition probability $P(\mathbf{x}(n+1)|\mathbf{x}(n))$ is a delta function $P(\mathbf{x}(n + 1)|\mathbf{x}(n)) = \delta^D(\mathbf{g}(\mathbf{x}(n + 1),\ \mathbf{x}(n),\ \mathbf{p}))$. With model errors, this is broadened, and with a Gaussian approximation to these model errors we write

$$\begin{aligned}
&P(\mathbf{x}(n+1)|\mathbf{x}(n)) \propto \\
&\quad \times \exp\left[-\frac{1}{2}\sum_{n=0,a=1}^{m-1,D} R_{fa}(g_a(\mathbf{x}(n+1),\ \mathbf{x}(n),\ \mathbf{p}))^2\right].
\end{aligned} \qquad (3)$$

  When the differential equation solver is explicit $\mathbf{g}(\mathbf{x}(n + 1),\ \mathbf{x}(n),\ \mathbf{p}) = \mathbf{x}(n + 1) - \mathbf{f}(\mathbf{x}(n),\ \mathbf{p})$.
- If the measurements are independent at different times, and the noise in each measurement is Gaussian, then the information transfer contribution to $A_0(\mathbf{X},\ \mathbf{Y})$ is proportional to

$$\frac{R_m}{2}\sum_{n=0,l=1}^{m,L} (y_l(n) - x_l(n))^2. \qquad (4)$$

These standard assumptions may be easily replaced within the context of the path integral, introducing little additional computational challenge using the Monte Carlo approach utilized below.

For these standard assumptions

$$\begin{aligned}A_0(\mathbf{X},\ \mathbf{Y}) &= \frac{R_m}{2}\sum_{n=0,l=1}^{m,L}(y_l(n)-x_l(n))^2\\ &+\frac{1}{2}\sum_{n=0,a=1}^{m-1,D}R_{fa}(g_a(\mathbf{x}(n+1),\ \mathbf{x}(n),\ \mathbf{p}))^2\\ &-\log[P(\mathbf{x}(0)].\end{aligned}\tag{5}$$

Further we take the initial distribution of states to reflect total ignorance of the initial state: $P(\mathbf{x}(0))$ is a uniform distribution. It then factors out of all calculations such as those in Eq. (1). Note that $A_0(\mathbf{X},\ \mathbf{Y})$ is *not* Gaussian in the state variables as the function $\mathbf{f}(\mathbf{x},\ \mathbf{p})$ is nonlinear in the $\mathbf{x}$.

The expected value of a state variable comes from $G(\mathbf{X})=\mathbf{X}$, and the marginal distribution of $x_c(n)$; $P_{c,n}(z)$ arises from $G(\mathbf{X})=\delta(z-x_c(n))$. From quantities such as this we can answer important sets of questions in the data assimilation process.

2.3 Monte Carlo evaluation

Using our formulation of the conditional probability density $P(\mathbf{X}|\mathbf{Y})$ for a path through the state space, we would like to evaluate approximations to quantities such as means, covariances about those means, and marginal distributions of parameters or state variables. These can be used to make estimates and predictions of quantities in the model, consistent with the observed data. Each of these quantities can be written as path integrals of the form given in Eq. (1), where $G(\mathbf{X})$ is chosen to be some interesting function of the path.

The numerical challenge is to evaluate these path integrals. One approach is to seek a stationary path where

$$\frac{\partial A_0(\mathbf{X},\ \mathbf{Y})}{\partial \mathbf{X}}=0,\tag{6}$$

and we have explored this in Toth et al. (2011) when the dynamics is deterministic, namely $R_f\to\infty$.

In working with data from experiments, the model is an incomplete representation of the underlying processes, and the deterministic variational method, where $R_f\to\infty$, imposes an equality constraint in the optimization that is likely to be too strong. The minimization of the action in Eq. (6) should be better as it embodies the notion of model errors. In practice, one can consider using Eq. (6) to select an initial path from which to begin the Monte Carlo procedure we outline here. Indeed, that is how we selected an initial guess for the path in the iterative algorithm described below.

An alternative to the variational approach is to generate a series of paths $\{\mathbf{X}^{(1)},\ldots,\mathbf{X}^{(N_{paths})}\}$ that are distributed in $(m+1)D$ dimensional path space according to $P(\mathbf{X}|\mathbf{Y})\propto\exp[-A_0(\mathbf{X},\ \mathbf{Y})]$. We can use these paths to approximate the distribution with

$$P(\mathbf{X}|\mathbf{Y})\approx\frac{1}{N_{\text{paths}}}\sum_{j=1}^{N_{\text{paths}}}\delta(\mathbf{X}-\mathbf{X}^{(j)}).$$

An estimate of the expected value of a function $\langle G(\mathbf{X})\rangle$ on the path follows as

$$\begin{aligned}\langle G(\mathbf{X})\rangle &= \int d\mathbf{X}\,G(\mathbf{X})P(\mathbf{X}|\mathbf{Y})\\ &\approx \tfrac{1}{N_{\text{paths}}}\textstyle\sum_{j=1}^{N_{\text{paths}}}G(\mathbf{X}^{(j)}).\end{aligned}\tag{7}$$

These paths can be thought of as representing the many possible time evolutions of the system state from an ensemble of initial conditions. They each evolve through path space according to the dynamics embodied in the transition probabilities entering the action $A_0(\mathbf{X})$ (we drop the explicit reference to the observations $\mathbf{Y}$ now). The parameters are considered as constants with respect to the dynamics, i.e., $dp(t)/dt=0$, and the distribution that is obtained reflects their influence on the state vector $\mathbf{x}(n)$ through the model equations $\mathbf{f}(\mathbf{x}(n),\ \mathbf{p})$.

We require a method which will produce paths distributed according to $\exp[-A_0(\mathbf{X})]$. There are several path integral Monte Carlo (PIMC) methods, such as Metropolis–Hastings or Hybrid Monte Carlo, that are designed to achieve this (Metropolis et al. 1953; Mackay 2003; Neal 1993). These methods make a biased random walk through the path (search) space that approaches the desired distribution.

2.4 Metropolis–Hastings Monte Carlo

One of the earliest developed methods for sampling from a high-dimensional distribution is the Metropolis–Hastings Monte Carlo method (Metropolis et al. 1953; Hastings 1970). This generates a sequence of paths $\mathbf{X}^{(0)},\ \mathbf{X}^{(1)},\ldots,\mathbf{X}^{(k)}$ from a random walk that is biased through an acceptance criterion that depends upon the distribution of interest. It is an example of a MCMC method because the sequence of paths may themselves be considered as states of a Markov Process. A Markov *chain* consists of a set of accepted paths, the set being indexed by the iterations k of the procedure.

The Metropolis–Hastings Monte Carlo method works by generating a new path $\mathbf{X}^{(k+1)}$ from the current path $\mathbf{X}^{(k)}$ in two steps. First, a candidate path $\mathbf{X}^{\text{proposed}}$ is proposed by adding an unbiased random displacement to the current path $\mathbf{X}^{\text{current}}$. The displacement may be to any subset of the components of $\mathbf{X}^{\text{current}}$; we restrict the distribution of this perturbation to be symmetric, assuring that the transition $\mathbf{X}^{\text{current}}\to\mathbf{X}^{\text{proposed}}$ is as likely as the reverse $\mathbf{X}^{\text{proposed}}\to\mathbf{X}^{\text{current}}$, to insure that the chain remains Markov.

Next, the proposed path is either accepted as the next path in the sequence $\mathbf{X}^{(k+1)} = \mathbf{X}^{\text{proposed}}$ or it is rejected so $\mathbf{X}^{(k+1)} = \mathbf{X}^{(k)}$. The probability for acceptance is

$$\min\left(1,\ \exp\left[-(A_0(\mathbf{X}^{\text{proposed}}) - A_0(\mathbf{X}^{\text{current}}))\right]\right). \tag{8}$$

This says that proposed changes that lower the action are accepted, while those that increase the action are accepted with probability $\exp[-\Delta A_0]$.

2.4.1 General procedure

An initial path $\mathbf{X}^{(0)}$ is supplied by the user and set to be the current path at iteration $k = 0$; the observed time series $\mathbf{Y}$ is data that is loaded in from a file. MCMC algorithms may take some time to converge to the correct distribution, and selecting an initial path that is close to the true solution generally leads to much faster convergence rates. In principle, the Metropolis–Hastings algorithm will eventually sample from the entire posterior distribution, but when distributions are multimodal, in practice the sampling chains will tend to stay in local minima corresponding to the initial guess. Our approach to this problem, which is common to all MCMC algorithms, is to use another optimization scheme (Toth et al. 2011) to perform a coarse search over broad parameter bounds and supply the result of this procedure as the starting point for the MCMC chains.

The MCMC calculation proceeds in two distinct phases; the first "burn-in" phase iterates this initial path guess while adjusting the size of the random perturbation α so that the acceptance rate of paths according to the Metropolis–Hastings criteria equation is approximately one-half. Adjusting the step size between iterations violates the Markovian requirement for symmetric transitions between the subsequent paths in the chain; however it also allows the chain to evolve more efficiently. Due to this violation, no statistics are gathered during the first phase.

During the second phase we uniformly sample from the chain to collect the constituent paths of our distribution. It is therefore important that α be held fixed during this period, at a value determined during the first phase. Other than that, the iterations proceed identically during both phases.

A single iteration consists of the following steps:

(1) If $k = 0$ evaluate $A_0(\mathbf{X}^{(0)})$.
(2) If $k > 0$, propose a change to component(s) of $\mathbf{X}$: $X_i^{\text{proposed}} \leftarrow X_i^{(k)} + U(-\alpha/2,\ \alpha/2) \cdot r_i$.
(3) Evaluate $A_0(\mathbf{X}^{\text{proposed}})$ and $\Delta A_0 = A_0(\mathbf{X}^{\text{proposed}}) - A_0(\mathbf{X}^{(k)})$.
(4) If $U(0,\ 1) < \min[1,\ \exp(-\Delta A_0)]$, then $\mathbf{X}^{(k+1)} \leftarrow \mathbf{X}^{\text{proposed}}$ (accept change).
Otherwise let $\mathbf{X}^{(k+1)} \leftarrow \mathbf{X}^{(k)}$ (reject change).

This loop is repeated until $k = k_{\text{final}}$, when sufficient statistics on the paths distributed as $\exp[-A_0(\mathbf{X})]$ have been collected. The subscript i refers to any component of the path $\mathbf{X}$, including individual time steps and the global parameters. r_i is a scale factor for perturbations to the ith component of $\mathbf{X}$.

The perturbation step size α is a dimensionless parameter between zero and one; it is set to be the same for all dimensions of $\mathbf{X}$, allowing any inter-dimensional scaling to be done by the user via the bounds. This is a simplification, not a requirement of the procedure. In this way, the actual perturbation to a component of $\mathbf{X}$ is given by a random number $U(-\alpha/2,\ \alpha/2)$ times the bounds-range (r_i), where U is a uniformly distributed random value between $-\alpha/2$ and $\alpha/2$.

2.4.2 Implementation for parallel architectures

The Metropolis–Hastings Monte Carlo method is simple and powerful, but it requires many path updates to achieve accurate statistics. One way to deal with this is to take advantage of parallel computing technology, using a GPU. With GPU technology it is possible to execute hundreds of threads of execution simultaneously on a single GPU. Typically each thread will perform the same operations, but on different pieces of the data. Because the paths are updated sequentially, the iteration steps cannot be parallelized in their entirety. However, during each iteration the path is perturbed in all $(m+1)D+K$ dimensions to give the candidate path $\mathbf{X}^{\text{proposed}}$, and $A_0(\mathbf{X}^{\text{proposed}})$ is calculated, both of which can be efficiently parallelized across the dimensions of $\mathbf{X}$.

All MCMC calculations reported here first generated 10^7 sample paths as a burn-in phase for the Metropolis–Hastings iterations. These were followed by a statistics collection phase of 10^8 iterations during which 10^3 accepted paths were uniformly sampled to create the approximate distribution $P(\mathbf{X}|\mathbf{Y})$. The data assimilation window comprised $m + 1 = 4,096$ points. The expression of the model error $\mathbf{g}(\mathbf{x}(n))$ is a discrete time version of deterministic equations for the neuron. We selected a fourth order Runge–Kutta integration scheme using a time step of $\Delta t = 0.01$ ms.

When the assimilation procedure was completed, a prediction using a fourth order Runge–Kutta scheme was performed on each of the accepted paths using state variables $\mathbf{x}(T)$ and the parameters associated with that path. The predicted trajectories were averaged to determine the expected value of $\mathbf{x}(t > T)$ and the standard deviation was evaluated about this mean. This gives the predicted quantities and their RMS errors as reported in the figures.

In order to assign values for R_f, the normalized deviation of the noise was estimated at 1 part in 10^3 for all dimensions of the model error. This normalized deviation was then scaled by the full range of the state variable and squared to get the variance for that dimension, so for $V \in [-200, 200]$, $R_f = 6.25$, while for the gating variables, $R_f = 10^6$.

We considered an experimental error of ±0.5 mV giving $R_m \approx 1$. We adjusted the Monte Carlo step size using a scaling factor α to achieve an acceptance rate near 0.5. The time required to perform each of our reported calculations with 10^8 candidate paths, each of dimension 16,402 (NaKL) to 20,498 (NaKLh), took about 10 h to complete on a single NVIDIA GTX 470 GPU. In our experience, provided that the dimension of the problem is roughly constant, the amount of time for a calculation scales roughly linearly with the number of GPUs.

In practice, as the Metropolis–Hastings procedure seeks paths distributed about the maxima of the probability distribution $\exp[-A_0(\mathbf{X})]$, a statistical minimization of $A_0(\mathbf{X})$ occurs when paths are accepted and rejected. This makes it a natural generalization of the procedure used in Toth et al. (2011) applicable to the situation where there is model error as well. As emphasized in this paper, the MCMC approach also results in expected errors for estimations and predictions.

2.5 Model neurons

2.5.1 NaKL model

The simplest HH model describes the dynamics of the voltage $V(t)$ across the membrane of a one compartment neuron containing two voltage-gated ion channels, Na and K, a passive leak current, and an electrode through which an external current $I_{\rm app}(t)$ can be applied. The model consists of an equation for voltage (Johnston and Wu 1995)

$$\begin{aligned}\frac{\mathrm{d}V(t)}{\mathrm{d}t} &= \frac{1}{C}\Big\{g_{\rm Na}m(t)^3h(t)(E_{\rm Na}-V(t))\\ &\quad +g_{\rm K}n(t)^4(E_{\rm K}-V(t))\\ &\quad +g_{\rm L}(E_{\rm L}-V(t))+I_{\rm DC}+I_{\rm app}(t)\Big\},\\ &= F_V(V(t), m(t), h(t), n(t))\end{aligned} \tag{9}$$

where the g terms indicate maximal conductances and the E terms reversal potentials, for each of the Na, K, and leak channels. $I_{\rm DC}$ is a DC current. Equations for the voltage dependent gating variables $a_i(t) = \{m(t), h(t), n(t)\}$ complete the model. We refer to this as the NaKL model.

Each gating variable $a_i(t) = \{m(t), h(t), n(t)\}$ satisfies a first order kinetic equation of the form

$$\frac{\mathrm{d}a_i(t)}{\mathrm{d}t} = \frac{a_{i0}(V(t)) - a_i(t)}{\tau_i(V(t))}. \tag{10}$$

The kinetic terms $a_{i0}(V)$ and $\tau_i(V)$ are taken here in the form

$$\begin{aligned}a_{i0}(V) &= \frac{1}{2}\left(1+\tanh\frac{(V-\mathrm{va})}{\mathrm{dva}}\right)\\ \tau_i(V) &= t_{a0}+t_{a1}\left(1-\tanh^2\frac{(V-\mathrm{vat})}{\mathrm{dvat}}\right)\\ &\quad +\frac{t_{a2}}{2}\left(1+\tanh\frac{(V-\mathrm{vatt})}{\mathrm{dvatt}}\right).\end{aligned} \tag{11}$$

The constants va, dva, . . . are selected to match the familiar forms for the kinetic coefficients usually given in terms of sigmoidal functions $(1 \pm \exp((V - V_1)/V_2))^{-1}$. As discussed in Part I, the tanh forms are numerically the same over the dynamic range of the neuron models but have better controlled derivatives when one goes out of that range during the required search procedures. This is less important here because the MCMC method does not use the derivatives, but we retain the same form for consistency.

In terms of the formalism of Sect. 2.1, the the model state variables are the $\mathbf{x}(t) = \{V(t), m(t), h(t), n(t)\}$, and the parameters are $\mathbf{p} = \{C, g_{\rm Na}, E_{\rm Na}, g_{\rm K}, E_{\rm K}, \ldots, \mathrm{dvatt}\}$. In a twin experiment, the data are generated by solving these HH equations for some initial condition $\mathbf{x}(0) = \{V(0), m(0), h(0), n(0)\}$ and some choice for the parameters, the DC current and $I_{\rm app}(t)$, the stimulating current. The measured data $y(t) = h(\mathbf{x}(t))$ is here just the voltage $V(t)$ produced by the model plus additive, independent noise at each time step drawn from a Gaussian distribution with zero mean and variance of 1 mV.

2.5.2 NaKLh model

Most neurons express a number of voltage-gated channels in addition to the sodium and potassium channels directly responsible for action potential generation (Graham 2002). These additional channels contribute to bursting, firing rate adaptation, and other behaviors. As in Toth et al. (2011), we are interested in whether the data assimilation procedure described here can be applied to more complex models than NaKL, and whether it can be used in the context of model specification to determine which channels should be included in the model. A model incorporating all the channels likely to be in a typical neuron is beyond the scope of this paper, but as a first step we extended the NaKL model to include the I_h current, which has moderately slower kinetics than the other channels in the model, and is activated by hyperpolarization (McCormick and Pape 1990). The I_h current was represented by an additional term in the HH voltage equation (9),

$$I_h(t) = g_h hc(t)(E_h - V(t)), \tag{12}$$

as well as an additional equation for the dynamics of the I_h gating variable:

$$\frac{\mathrm{d}hc(t)}{\mathrm{d}t} = \frac{hc_0(V(t)) - hc(t)}{\tau_{hc}(V)}$$

$$hc_0(V) = \frac{1}{2}\left(1 + \tanh\left[\frac{(V - \mathrm{vhc})}{\mathrm{dvhc}}\right]\right)$$
$$\tau_{hc}(V) = t_{hc0} + t_{hc1}\tanh\left[\frac{(V - \mathrm{vhct})}{\mathrm{dvhct}}\right]. \quad (13)$$

3 Results

3.1 NaKL model parameters estimated from NaKL data

We begin with an examination of the PIMC data assimilation procedure using the NaKL model. We selected a set of parameters (Table 1, "Data") using standard textbook values for maximal conductances ($g_{\mathrm{Na}}, g_{\mathrm{K}}, g_{\mathrm{L}}$) and reversal potentials ($E_{\mathrm{Na}}, E_{\mathrm{K}}, E_{\mathrm{L}}$). The parameters in the kinetic equations for the gating variables $\{m(t), h(t), n(t)\}$ came from a fit to standard expressions for the time constants $\tau_i(V)$ and driving functions $a_{i0}(V)$ using the hyperbolic tangent functions in Eq. (11). Choosing appropriate initial conditions, we integrated the NaKL HH equations with an input current $I_{\mathrm{app}}(t)$ consisting of a scaled waveform taken from the solution to a chaotic dynamical system. The amplitude of the waveform was selected so it depolarized and hyperpolarized the model neuron, evoking action potentials and traversing the biologically realistic regions of the neuron's state space. Based on our findings in Toth et al. (2011), the frequency of $I_{\mathrm{app}}(t)$ was slow relative to the response rate of the neuron.

Table 1 Parameter values in simulated data from NaKL HH model and estimates from the PIMC algorithm

Parameter	'Data'	Estimate	SD	Units
g_{Na}	120.0	108.6	7.3	mS/cm^2
E_{Na}	55.0	55.8	0.70	mV
g_{K}	20.0	18.2	1.0	mS/cm^2
E_{K}	−77.0	−78.0	0.67	mV
g_{L}	0.3	0.316	0.021	mS/cm^2
E_{L}	−54.4	−55.3 mV	1.5	mV
vm=vmt	−34.0	−36.5	1.5	mV
dvm=dvmt	34.0	35.6	0.77	mV
t_{m0}	0.01	0.136	0.096	ms
t_{m1}	0.5	0.407	0.087	ms
vh=vht	−60.0	−62.1	1.8	mV
dvh=dvht	−19.0	−23.6	2.0	mV
t_{h0}	0.2	0.157	0.071	ms
t_{h1}	8.5	8.3	0.16	ms
vn=vnt	−65.0	−64.2	1.3	mV
dvn=dvnt	45.0	44.6	0.40	mV
t_{n0}	0.8	0.35	0.24	ms
t_{n1}	5.0	4.84	0.086	ms

The SD column indicates the standard deviation of the posterior distribution

To construct $A_0(\mathbf{X}, \mathbf{Y})$ we took $m+1 = 4,096$ data points with $\Delta t = 0.01$ ms writing $\mathbf{g}(\mathbf{x}(n), \mathbf{x}(n+1), \mathbf{p}) = \mathbf{x}(n+1) - \Delta t\mathbf{f}(\mathbf{x}(n), \mathbf{p})$ where $\mathbf{f}(\mathbf{x}(n), \mathbf{p})$ is represented as an explicit 4th-order Runge–Kutta integration scheme. Using the methods described in Sect. 2.3, we evaluated expected values for the state variables $\langle x_a(n)\rangle$ and parameters through the observation period, and also evaluated second moments to yield standard deviations about these expected values. The dimension of the integral we are approximating is 4 (4,095) + 18 = 16,398.

Using the model with these parameters and state variables at T as initial conditions, we predict forward from the end of the observation/assimilation time interval, also using 4th-order Runge–Kutta integration. In the data assimilation procedure we have many accepted paths distributed as $\exp[-A_0(\mathbf{X})]$. We predict into $t > T$ using $\mathbf{x}(T)$ for each path and the parameters estimated with that accepted path. This permits us to calculate a mean and standard deviation for each state variable as the system continues to evolve.

Figure 1a shows the estimated membrane voltage (red dots) ± standard deviation (green band) overlaid on the true voltage (black line) for data generated with the NaKL model. The data assimilation window consists of the time points before the vertical blue line. Time points after the blue line compare the predicted response of the model to the true voltage after time T. Figure 1b compares the estimated and predicted values for the Na$^+$ activation variable ($m(t)$), an unobserved state variable, with the known values.

The accuracy with which the path integral estimates track the observed and unobserved states through the observation window is clear in Fig. 1. In the prediction window the expected voltages and $m(t)$ give quite good values for the times when spikes occur, indicating that the Na and K currents are well represented when an action potential is generated. The dynamical response in the regions of hyperpolarized response is less accurate, though all estimates lie within one standard deviation of the expected value. As in Toth et al. (2011) we could take this as a signal that the stimulus current did not spend enough time in the hyperpolarized voltage region to stimulate the dynamics there very well.

Table 1 compares the parameter values used to simulate the data from the NaKL model with the estimates (± standard deviation) made from the noisy voltage alone. For all but t_{m0} and t_{n0} the expected values are almost identical to the true values.

3.1.1 Bias in the conditional expected values arising from model error

It is noteworthy that while the posterior errors are small in the estimates in Table 1 (and later tables of parameter estimates) the expected or mean value appears to be biased away from the known value. This bias comes from our procedure and

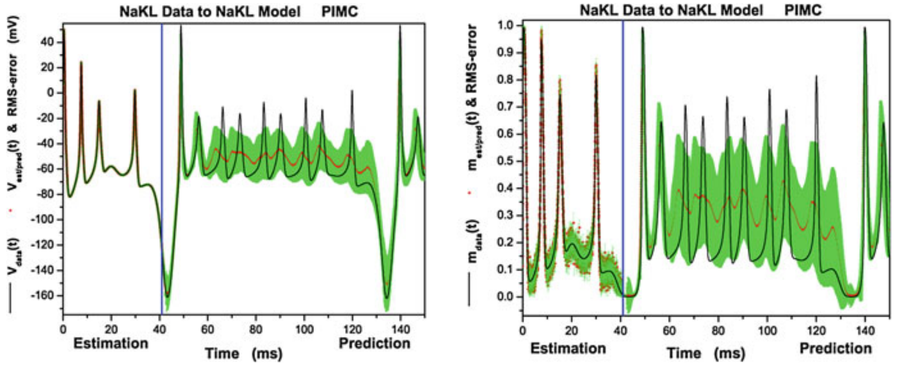

Fig. 1 NaKL model. **a** Comparison of the membrane voltage and the estimates and predictions. In the assimilation window ($t < 40.95$ ms, *blue line*), the expected value of the voltage conditioned on the observations (V_{est}) is shown by *red dots*, with the standard deviation of this distribution shown in *green*. For $t \geq 40.95$ ms, the *red* and *green symbols* show the mean and standard deviation of the distribution of responses predicted forward in time using the estimates of the parameters and the state variables at $T = 40.95$ ms. The membrane voltage is shown as a *black line*. **b** Comparison of the known value of the Na$^+$ activation variable (*black trace*) and the distribution of the estimates (*red* ± *green*; prior to *blue line*) and predictions from the model (*red* ± *green*; after *blue line*). (Color figure online)

is associated with having model error as part of the action $A_0(\mathbf{X})$.

The distribution of paths $\exp[-A_0(\mathbf{X})]$ is the solution to a Fokker–Planck equation of the form

$$\frac{\mathrm{d}\mathbf{X}(s)}{\mathrm{d}s} = -\frac{\partial A_0(\mathbf{X}(s))}{\partial \mathbf{X}(s)} + \sqrt{2}\,N(0,\ 1), \tag{14}$$

where $\mathbf{X}(s)$ is the state space path as a function of "time" s, and $N(0,\ 1)$ is Gaussian while noise with zero mean and variance unity. An equivalent to our Metropolis–Hastings Monte Carlo procedure is to solve this stochastic differential equation in $(m+1)D$-dimensions, where one can show that as $s \to \infty$, the distribution of paths is precisely $\exp[-A_0(\mathbf{X})]$. The Monte Carlo algorithm is seen as a search for minima of the action along with accounting for the fluctuations about the minima. All of this is guided by the observations as they enter the action.

To demonstrate the point about biases in the estimation, suppose we had two measurements $y_1,\ y_2$ and two model outputs with the model taken as linear $x_2 = Mx_1$. Then the action we would associate with this, including model error, is

$$A(x_1,\ x_2) = \frac{1}{2}\left[(y_1-x_1)^2+(y_2-x_2)^2 + R(x_2-Mx_1)^2\right], \tag{15}$$

and this has its minimum at

$$\begin{aligned} x_1 &= \tfrac{(1+R)y_1+RMy_2}{1+R(1+M^2)} \\ x_2 &= \tfrac{(1+RM^2)y_2+RMy_1}{1+R(1+M^2)}. \end{aligned} \tag{16}$$

This clearly show the bias we anticipated. As $R \to \infty$, we see that the bias remains, but $x_2 = Mx_1$ is enforced. If $R = 0$, however, the minimum is at $x_1 = y_1$, $x_2 = y_2$, and if the dynamics underlying the data source satisfies $y_2 = My_1$, the same holds for the model.

3.2 NaKLh model parameters estimated from NaKLh data

To increase the complexity of the model, an additional voltage-gated current was added (I_h; see Sect. 2.5). As in the previous section, data were simulated from known parameters and initial conditions (Table 2, 'Data'). Based on our findings in Toth et al. (2011), we used a strong stimulus current to ensure that the I_h current was sufficiently activated. Then, using the noisy voltage output from the simulation, the PIMC algorithm was used to estimate the parameters and unobserved states of the model. These estimates are as good or better than for the simpler NaKL model (Table 2). Moreover, the additional $4,096+8$ dimensions added to the integral did not increase the posterior variance or substantially decrease the speed of the calculation.

Figure 2 compares the known values for $V(t)$ and the unobserved gating variable $hc(t)$ of the newly added I_h current against the estimates and their posterior error. As with the NaKL model, the estimates of voltage (A) and the I_h gating variable (B) closely follow the true values during the data assimilation window and beyond.

3.3 NaKLh model parameters estimated from NaKL data

A critical consideration in applying this method to experimentally obtained data is model selection. If the goal is to make inferences about biologically relevant properties such as the set of channels expressed by an individual neuron, then we need some method of determining which channels contribute to a neuron's electrophysiological behavior and should be included in the model. One approach

Table 2 Parameter values in simulated data from NaKLh HH model and estimates from the PIMC algorithm

Parameter	‘Data’	Estimate	SD	Units
g_{Na}	120.0	116.0	2.8	mS/cm^2
E_{Na}	55.0	55.17	0.19	mV
g_{K}	20.0	19.40	0.20	mS/cm^2
E_{K}	−77.0	−77.38	0.27	mV
g_{L}	0.30	0.2987	0.0098	mS/cm^2
E_{L}	−54.4	−56.8	1.8	mV
vm=vmt	−34.0	−33.64	1.2	mV
dvm=dvmt	34.0	36.58	1.15	mV
t_{m0}	0.01	0.051	0.023	ms
t_{m1}	0.5	0.45	0.060	ms
vh=vht	−60.0	−67.29	3.2	mV
dvh=dvht	−19.0	−18.9	1.1	mV
t_{h0}	0.2	0.34	0.14	ms
t_{h1}	8.5	6.25	2.5	ms
vn=vnt	−65.0	−68.2	2.6	mV
dvn=dvnt	45.0	44.6	0.63	mV
t_{n0}	0.8	0.97	0.12	ms
t_{n1}	5.0	4.8	0.38	ms
g_h	1.21	1.18	0.029	mS/cm^2
E_h	40.0	37.8	0.65	mV
vhc	−75.0	−77.7	0.64	mV
$dvhc$	−11.0	−11.1	0.26	mV
t_{hc0}	0.1	0.118	0.043	ms
t_{hc1}	193.5	177.24	3.1	ms
$vhct$	−80.0	−81.3	0.67	mV
$dvhct$	21.0	21.4	0.26	mV

The SD column indicates the standard deviation of the posterior distribution

is to start with a relatively complex model that includes all of the currents likely to be in the neuron of interest (possibly using genetic expression data to restrict the list of candidates). All such currents would have the HH form

$$I_{\text{current}}(t) = g_{\text{current}}(mc(t))^q(hc(t))^p(E_{\text{current}} - V(t)), \quad (17)$$

where $mc(t)$ is a state variable associated with transitions to the open state, $hc(t)$ is a state variable associated with transitions to closed states, and p and q are integers. Differential equations for the kinetics of the state variables would need to be specified as well. If this term is included in the model but g_{current} is not distinguishable from zero, or if the other parameters are nonsensical, then we could infer that the current does not contribute appreciably to the neuron’s behavior.

Using the variational approximation introduced in Part I, we found that if we estimated parameters of an NaKLh model from data that were generated from an NaKL model, the estimated conductance for the missing I_h current was extremely close to zero, supporting this approach. We repeated this experiment using the numerical approximation to the full integral and obtained the same result. NaKL data were simulated by using an NaKLh model in which g_h was fixed at zero. Parameters associated with the I_h current were left at their estimated values, though they had no effect on the voltage behavior of the neuron. The responses of this model were identical to the NaKL model in which no I_h current was specified. Table 3 compares values for the parameters used to simulate the data with the values estimated using the PIMC method. The maximal conductance g_h is small, indicating that I_h does not contribute to the responses of the neuron. We note that the estimate of g_h in Toth et al. (2011) is much smaller, but in that experiment we did not include observational noise. Reducing the observational noise in this experiment leads to a corresponding decrease in the g_h estimate (not shown).

As important as the absolute value of the estimated conductance is the posterior error associated with the estimate, which we are able to estimate using the PIMC method.

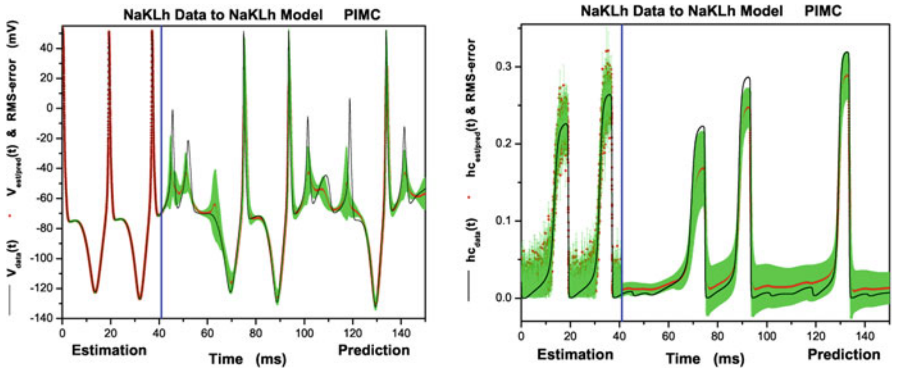

Fig. 2 NaKLh model. **a** Comparison of the membrane voltage with estimates during the assimilation window ($t < 40.95$ ms) and predictions from the model ($t \geq 40.95$ ms). As in Fig. 1, the known values are indicated by the *black trace*, and the mean and standard deviation of the posterior distributions are indicated by *red dots* and *green bars*, respectively. **b** Comparison of the known values for the I_h activation gating variable $hc(t)$ with the estimates and predictions from the model. (Color figure online)

Table 3 Parameter values in simulated data from NaKLh HH model and estimates from the PIMC algorithm

Parameter	'Data'	Estimate	SD	Units
g_{Na}	120	114.1	2.2	mS/cm^2
E_{Na}	55.0	55.2	0.12	mV
g_{K}	20.0	19.9	0.19	mS/cm^2
E_{K}	−77.0	−77.1	0.33	mV
g_{L}	0.30	0.292	0.0059	mS/cm^2
E_{L}	−54.4	−55.3	0.51	mV
vm=vmt	−34.0	−31.1	1.7	mV
dvm=dvmt	34.0	34.0	1.2	mV
t_{m0}	0.01	0.071	0.030	ms
t_{m1}	0.5	0.58	0.025	ms
vh=vht	−60.0	−52.2	2.7	mV
dvh=dvht	−19.0	−20.1	0.66	mV
t_{h0}	0.2	1.1	0.42	ms
t_{h1}	8.5	8.8	2.3	ms
vn=vnt	−65.0	−63.9	1.2	mV
dvn=dvnt	45.0	43.8	1.5	mV
t_{n0}	0.8	0.88	0.083	ms
t_{n1}	5.0	5.3	0.15	ms
g_{h}	0.0	0.019	0.015	mS/cm^2
E_h	−40.0	−27.0	1.3	mV
vhc	−75.0	7.4	1.1	mV
$dvhc$	−11.0	−43.5	0.55	mV
t_{hc0}	0.1	2.8	0.057	ms
t_{hc1}	193.5	207.6	1.9	ms
$vhct$	−80	−99.6	0.27	mV
$dvhct$	21.0	56.0	0.42	mV

$g_h = 0.0$ in the simulated data, indicating the absence of this channel

In contrast to the other parameters where the posterior error is a small fraction of the expected value, the error for g_h is nearly as large as the expected value. The estimates for the I_h kinetics are wrong, but can be ignored because $g_h \approx 0$. If the current does not contribute to the data, estimates of its properties are unreliable. The idea that we could build a "large" model of all neurons and use the data to prune off currents that are absent is plausible and supported by this calculation. Whether this optimistic viewpoint will persist as we confront laboratory data with our methods is yet to be seen.

Figure 3 compares the measured voltage and K$^+$ activation variable against the estimated values during the observation window (0–40.95 ms). The expected values follow the true values closely, with small posterior errors, indicating that the presence of the I_h current in the estimation model does not negatively impact the estimation procedure even though this current is not present in the data. More importantly, the predicted voltage obtained by integrating forward using the estimated parameters and state variables at $T = 40.95$ ms closely follows the true voltage. The quality of the prediction would support the inference, based on the voltage data alone, that I_h does not contribute substantially to the behavior of this neuron.

3.4 NaKL model parameters estimated from NaKLh data

The complementary approach to the model selection problem is to build up from a simple model, adding complexity to address aspects of the data that are not fit well. To address the feasibility of this approach, we used data simulated from the NaKLh model to fit parameters from the NaKL model, which is missing key parameters and state variables. Figure 4a shows a comparison of the estimated and predicted voltage from the NaKL model with the true values from the NaKLh model. In the observation window there is no indication that there is anything wrong with the model, but the dramatic failure to predict the response of the neuron after the observation window leaves no doubt that something is not consistent.

Although one cannot do this easily in a laboratory experiment, in a twin experiment we are able to examine how well

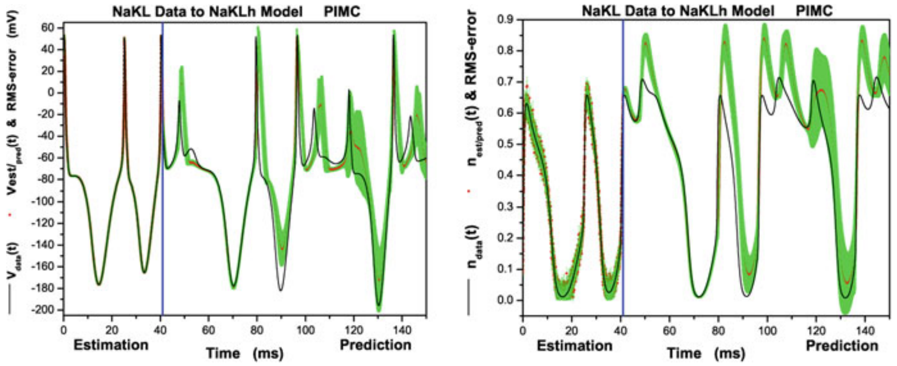

Fig. 3 NaKLh model parameters estimated from NaKL data. **a** Comparison of the membrane voltage with estimates during the assimilation window ($t < 40.95$ ms) and predictions from the model ($t \geq 40.95$ ms). This *panel* is almost identical to Fig. 2a, except that the I_h current does not contribute to the simulated data. **b** Comparison of the known values for the K$^+$ inactivation gating variable $n(t)$ with the estimates and predictions from the model

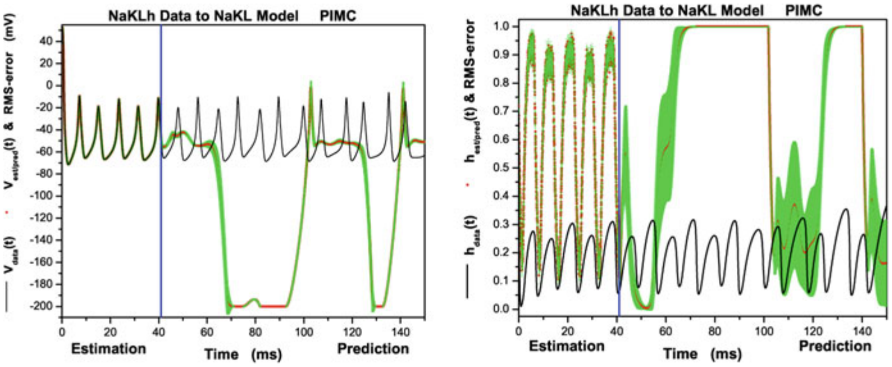

Fig. 4 NaKL model parameters estimated from NaKLh data. **a** Comparison of the known membrane voltage with estimates during the assimilation window ($t < 40.95$ ms) and predictions from the model ($t \geq 40.95$ ms). This *panel* is almost identical to Fig. 2a, except that model does not include the I_h current that is present in the simulated data. **b** Comparison of the known values for the Na^+ inactivation gating variable $h(t)$ with the estimates and predictions from the model

the estimations behave when estimating the gating variables. Figure 4b shows the estimated Na^+ inactivation variable $h(t)$ and its known value from our knowledge of the full state of the observed neuron model. It is clear that the estimates of the unobserved variable during the observation window fails to reflect the underlying system. One interpretation is that because the model is too simple to represent the actual system, the estimated parameters and state variables are driven to unrealistic values, which is revealed when the model is used to predict novel data. Although the failure of the NaKL model to capture the NaKLh behavior does not give any direct indication of what is missing, it does provide a clear basis for comparing families of models to determine which one best represents the underlying process.

4 Discussion

The equations describing the dynamics of a broad range of voltage-gated ion channels and their effect on the membrane voltage of neurons have been known for many years, but due to the large number of channels expressed in nature and the overall nonlinearity of neuronal systems it has not been possible to use recordings of membrane voltage alone to determine what channels are present and their kinetic parameters. The problem is a general one of finding the paths through a model state space that are consistent with observations of some subset of the state variables (here, voltage), as well as with the internal dynamics of the model.

When the measurements are noisy, the model has errors, and the state of the model is uncertain when observations commence, this is a problem in statistical physics. We presented an exact formulation of the path integral that describes this problem previously (Toth et al. 2011; Abarbanel 2009), along with a variational approximation based on a saddle path estimation. We found that this variational method provided accurate estimates of channel densities and kinetic parameters when applied to simulated data, and that the estimation procedure was robust to additive noise as well as errors in the model specification.

We have extended our earlier work here with effective numerical approximations to the full path integral, using a Metropolis–Hastings Monte Carlo technique. The chief advantage of this approach is that it allows one to find not only the optimal path through the state space (and the associated parameters), but to sample from the joint probability distribution of the paths and parameters, conditioned on the data observations. The variance of this posterior distribution provides valuable information about the degree to which the observed data constrains the model, and can indicate when additional data may be needed. It also allows generation of posterior predictive distributions—the expected behavior given the model and the observed data—which are useful in model validation and selection. In other words, the numerical approximation to the full integral not only allows transfer of information from the data to the model, but also indicates how much information was transferred.

Interestingly the Metropolis–Hastings method, as with other approaches (Andrieu et al. 2010), seeks paths near the maxima of the distribution $\exp[-A_0(\mathbf{X})]$ so that it represents, in effect, a statistical version of the stationary path method discussed in Toth et al. (2011). It has the computational advantage of not requiring any derivatives of the action, so if the model contains thresholds or “switches” there is no problem in the PIMC method.

Using this method, we repeated several of the numerical experiments with simulated data described earlier (Toth et al. 2011) in order to show that estimates are accurate and provide good forward predictions for simple HH-type biophysical models. The posterior variance of these estimates was small, indicating that a small amount of data (41 ms) was sufficient to provide a high degree of confidence in the estimates, and furthermore, that the behavior of the models was strongly dependent on the parameter values.

We also demonstrated, by estimating model states and parameters with the wrong complement of channels, one too many or one too few, that we could easily identify these situations. In the case where the model contained a channel that was not present in the data, the estimated conductance for that additional channel was small and the posterior uncertainty was large, allowing the "extra" channel to be pruned from the model with confidence. When the model contained too few channels the forward predictions were highly inaccurate, indicating that the action of the missing channels plays an important dynamical role. From these observations, it appears that in dealing with preparations where the full complement of channels is not known, it is preferable to start with a larger model that includes any channel with a reasonable probability of being present, which might be informed by the known biology of the system such as prior pharmacological experiments or genetic expression patterns, and then remove channels for which the maximum conductance is estimated to be close to zero or has a high error. We recommend this as a general strategy for selecting good models when experimental data is used.

We noted, as exhibited by our estimates in Tables 1, 2, 3, that because the estimation procedure is seeking the minima of $A_0(\mathbf{X})$ along with the fluctuations about those minima, and since the action contains terms representing the error in the models, the expected values of the estimates will be biased, even when the posterior error about the expected values is small. This bias is discernable in the twin experiments we discuss here, but would not be known in the application of our methods to laboratory data. It is important to be cognizant of the bias, however.

From a practical standpoint, the major disadvantage in using the full path integral rather than the variational approximation is that it can require much more computational time. This is largely ameliorated by the parallel GPU algorithm we utilized, and the low cost of using a GPU for the computation. Furthermore, given that both methods provide similar levels of accuracy, the variational method can be used for exploratory analysis, followed by a subsequent in-depth analysis of the full distribution. We have used the variational principle, whether implemented through IPOPT or via another optimization routine, as a source of the first guess for a path in the PIMC method. This appears to provide a much better starting path than a random guess—better, in the sense that the PIMC method converges in fewer iterations to a distribution with good expectation values. Other strategies are explored by Andrieu et al. (2010).

There has been substantial interest in recent years in using noisy measurements of neurons to infer biologically relevant properties, using a variety of optimization methods (Druckmann et al. 2007; Abarbanel 2009; Huys and Paninski 2009; Lepora et al. 2011). The ability to make such inferences opens possibilities of using very brief intracellular recordings to closely characterize individual neurons, to reveal the distribution of various biological properties over large populations of neurons, or to track changes in these properties through learning or changes in behavioral state. The methods we describe here and in Part I have several new advantages. They provide estimates not only of the maximal conductances of fixed channel types, but also of the parameters governing the gating kinetics of unknown channels. The Monte Carlo method also provides information about the posterior uncertainty in the parameter estimates. These features may be of particular value in analyzing systems where the candidate channels are not well known.

The experiments we describe here necessarily focused on relatively simple and generalizable model neurons. Extending the method to more complex systems will require the incorporation of additional knowledge about the types of channels most likely to be present or the anatomy of the neurons. Dendritic dynamics are not very important in responses to injected current, but will become so when we consider synaptic inputs and network dynamics (Johnston and Narayanan 2008). Twin experiments like the ones we have presented here will continue to play an important role in studying biological data: no matter how complex the model becomes, twin experiments can be used to generate simulated data to determine what conditions are necessary for obtaining a good estimate of the parameters, which in turn can be used to optimize stimulation protocols and other aspects of experimental design.

Acknowledgements Support from the US Department of Energy (Grant DE-SC0002349) and the National Science Foundation (Grants IOS-0905076, IOS-0905030, and PHY-0961153) are gratefully acknowledged. Partial support from the NSF sponsored Center for Theoretical Biological Physics is also appreciated. Discussions with Jack Quinn on GPU computing were very valuable in our numerical work reported here. He provided us with the GPU computing strategy we have employed.

References

Abarbanel HD (2009) Effective actions for statistical data assimilation. Phys Lett A 373(44):4044–4048

Andrieu C, Doucet A, Holenstein R (2010) Particle Markov chain Monte Carlo methods. J R Stat Soc B 72(3): 269–342. doi:10.1111/j.1467-9868.2009.00736.x

Druckmann S, Banitt Y, Gidon A, Schürmann F, Markram H, Segev I (2007) A novel multiple objective optimization framework for constraining conductance-based neuron models by experimental data. Front Neurosci 1(1): 7–18. doi:10.3389/neuro.01.1.1.001.2007

Evensen G (2009) Data assimilation: the ensemble Kalman filter. 2nd edn. Springer, Berlin

Graham L (2002) In: . Arbib MA (ed) The handbook for brain theory and neural networks. 2nd edn. MIT Press, Cambridge pp 164–170

Hastings WK (1970) Monte Carlo sampling methods using Markov chains and their applications. Biometrika 57(1): 97–109. doi:10.1093/biomet/57.1.97

Huys QJM, Paninski L (2009) Smoothing of, and parameter estimation from, noisy biophysical recordings. PLoS Comput Biol 5(5):e1000379. doi:10.1371/journal.pcbi.1000379

Johnston D, Narayanan R (2008) Active dendrites: colorful wings of the mysterious butterflies. Trends Neurosci 31(6): 309–316. doi:10.1016/j.tins.2008.03.004

Johnston D, Wu SMS (1995) Foundations of cellular neurophysiology. MIT Press, Cambridge

Lepora NF, Overton PG, Gurney K (2011) Efficient fitting of conductance-based model neurons from somatic current clamp. J Comput Neurosci. doi:10.1007/s10827-011-0331-2

Mackay DJC (2003) Information theory, inference and learning algorithms. Cambridge University Press, Cambridge

McCormick DA, Pape HC (1990) Properties of a hyperpolarization-activated cation current and its role in rhythmic oscillation in thalamic relay neurones. J Physiol 431(1):291–318

Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E (1953) Equation of state calculations by fast computing machines. J Chem Phys 21: 1087–1092. doi:10.1063/1.1699114

Neal RM (1993) Probabilistic inference using Markov chain Monte Carlo methods. Tech. Rep. CRG-TR-93-1, University of Toronto

Toth BA, Kostuk M, Meliza CD, Abarbanel HDI, Margoliash D (2011) Dynamical estimation of neuron and network properties I: variational methods. Biol Cybern 105(3):217–237