ADAPTIVE DISCRETE STOCHASTIC OPTIMIZATION ALGORITHM FOR LEARNING NERNST POTENTIAL IN NERVE CELL MEMBRANE ION CHANNELS

Vikram Krishnamurthy

Electrical and Computer Engineering,
University of British Columbia,
Vancouver, V6T 1Z4, Canada
email: vikramk@ece.ubc.ca

Shin-Ho Chung

Biophysics Group,
Research School of Physical Sciences,
Australian National University,
ACT 0200, Australia.
email: shin-ho.chung@anu.edu.au

ABSTRACT

We present discrete stochastic optimization algorithms that adaptively learn the Nernst potential in membrane ion channels. The proposed algorithms dynamically control both the ion channel experiment and the resulting Hidden Markov Model (HMM) signal processor and can adapt to time-varying behaviour of ion channels. One of the most important properties of the proposed algorithms are their its self-learning capability – they spends most of the computational effort at the global optimizer (Nernst potential).

1. INTRODUCTION

An ion channel is a hole or pore in a nerve cell membrane. In physical structure, an ion channel is a large protein molecule whose different configurations correspond to the ion channel being in a *closed* state or *open* state. The measurement of ionic currents flowing through single ion channels in cell membranes has been made possible by the giga-seal *patch-clamp* technique [1]. This was a major breakthrough for which the authors of [1] won the 1991 Nobel prize in Medicine. Because all electrical activities in the nervous system, including communications between cells and the influence of hormones and drugs on cell function, are regulated by membrane ion channels, understanding their mechanisms at a molecular level is a fundamental problem in biology. Ion channel currents are typically of the order of pico-amps (i.e. , 10^{-12} amps). In patch clamp experiments these minute ion channel currents are obfuscated by large amounts of thermal noise. Chung *et al.* [2] first introduced the powerful paradigm of Hidden Markov Models (HMMs) to characterize patch-clamp recordings of small ion channel currents contaminated by random and deterministic noise. By using sophisticated HMM signal processing methods, [2] demonstrated that the underlying parameters of the HMM could be obtained to a remarkable precision despite the extremely poor signal to noise ratio.

In this paper, we address the deeper and more fundamental problem of how to adaptively learn and control the behaviour (open state current level) of a single ion channel in a nerve cell membrane. By using recent state-of-the art methods from the electrical engineering disciplines of discrete-event-systems and stochastic control, we develop algorithms to adaptively control the applied voltage to a patch clamp experiment in order to dynamically learn the so called "*Nernst*" potential and current-voltage characteristics of the ion channel.

I-V Curve and Nernst Potential: Ion channel current measurements from a patch clamp experiment (after suitable anti-aliasing filtering and sampling) shows that the channel current is piecewise constant discrete time signal that randomly jumps between two values – zero amperes which denotes the *closed state* of the channel, and $I(v)$ amperes which denotes the *open* state. $I(v)$ is called the *open-state* current level. The open state current level $I(v)$ depends on the voltage v that is applied by the experimenter to the ion channel. Let $\{i_n(v)\}$ denote the discrete-time ion channel current sequence. In characterizing different types of ion channels, neurobiologists routinely construct current-voltage (I-V) curves. The curve represents the variation of the open state current level $I(v)$ of the ion channel as a function of the applied voltage value v. Such I-V curves yield a unique signature of a particular ion channel, revealing its operating characteristics. The zero point of the I-V curve, i.e. , the voltage v^* at which the open state current level $I(v^*)$ is zero, is known as the *Nernst potential*. The Nernst potential gives information about the relative concentrations at the two faces of the ionic channel. The value of the open state current level $I(v)$ is described by the Nernst-Planck equation that combines Ohm's and Fick's laws.

Our approach: The aim of this paper is to propose algorithms that efficiently learns the I-V curve from the noisy

This work was supported by NSERC and British Columbia Advanced Systems Institute

observed channel current sequence $\{y_n(v)\}$ by dynamically controlling the applied voltage v. The proposed algorithm dynamically controls (schedules) the choice of voltage v at which the ion channel operates in order to efficiently estimate the Nernst potential (zero current point) and deduce how the current increases or decreases as the applied voltage deviates from the Nernst potential. Thus at a given time instant, given the current estimate from a HMM estimator operating at a particular voltage, the aim is to devise a scheduling algorithm that dynamically decides which voltage value to pick at the next time instant to apply to the ion channel. The most important aspect of the resulting combined experiment scheduling/HMM estimation algorithm is its self learning capability – it is provably convergent to the Nernst potential estimate and is provably efficient – that is the algorithm spends more time running the ion channel at the Nernst potential than any other voltage. These learning algorithms are based on recent discrete stochastic approximation algorithms that have recently been developed in the operations research literature [3, 4].

2. ION CHANNEL MODEL

Since our aim in this paper is to adaptively learn and control the behaviour of an ion channel current modelled as a HMM, in this subsection we formally define the HMM for the ion channel current and briefly describe MLE algorithms for HMMs. Such probabilistic models for ion channels based on HMMs are now widely used [2, 5].

Markov Model for Ion Channel Current: Suppose a patch clamp experiment is conducted with a voltage v applied across the ion channel. Then, as described in [2, 5], the ion channel current $\{i_n(v)\}$, can be modelled as a three state homogeneous first order Markov chain. The state space of this Markov chain is $\{0_{\rm g}, 0_{\rm b}, I(v)\}$ corresponding to the physical states of *gap mode*, *burst-mode-closed* and *burst-mode-open*. For convenience, we will refer to the burst mode closed and burst-mode-open states as the *open* and *closed* states, respectively. In the gap mode and the closed state the ion channel current is zero. In the open state, the ion channel current has a value of $I(v)$.

The (3×3) transition probability matrix $A(v)$ of the Markov chain $\{i_n(v)\}$, which governs the probabilistic behaviour of the channel current, is given by

$$A(v) = \begin{array}{c|ccc} & 0_{\rm g} & 0_{\rm b} & I(v) \\ \hline 0_{\rm g} & a_{11}(v) & a_{12}(v) & 0 \\ 0_{\rm b} & a_{21}(v) & a_{22}(v) & a_{23}(v) \\ I(v) & 0 & a_{32}(v) & a_{33}(v) \end{array} \qquad (1)$$

The elements of $A(v)$ are the transition probabilities $a_{ij}(v) = P(i_{n+1}(v) = j | i_n(v) = i)$ where $i, j \in \{0_{\rm g}, 0_{\rm b}, I(v)\}$. The zero probabilities in the above matrix $A(v)$ reflect the fact that a ion channel current cannot directly jump from the gap mode to the open state, similarly an ion channel current cannot jump from the open state to the gap mode.

HMM Observations: Let $\{y_n(v)\}$ denote the measured noisy ion channel current at the electrode when conducting a patch clamp experiment:

$$y_n(v) = i_n(v) + w_n(v), \quad n = 1, 2, \ldots \qquad (2)$$

Here $\{w_n(v)\}$ is thermal noise and is modelled as zero mean white Gaussian noise with variance $\sigma^2(v)$. Thus the observation process $\{y_n(v)\}$ is a Hidden Markov model sequence parameterized by the model $\lambda(v) = \{A(v), I(v), \sigma^2(v)\}$ where v denotes the applied voltage.

HMM Parameter Estimation of Current Level $I(v)$: Given the HMM mode for the ion channel current above, estimating $I(v)$ for a fixed voltage v, involves processing the noisy observation $\{y_n(v)\}$ through a Hidden Markov Model maximum likelihood parameter estimator. The most popular way of computing the maximum likelihood estimate (MLE) $I(v)$ is via the Expectation Maximization (EM) algorithm (Baum Welch equations). Let $\hat{I}_\Delta(v)$ denote MLE of $I(v)$ based on the Δ-point measured channel current sequence $(y_1(v), \ldots, y_\Delta(v))$. For sufficiently large batch size Δ due to the asymptotic normality of the MLE for a HMM [6], $\sqrt{\Delta}\left(\hat{I}_\Delta(v) - I(v)\right) \sim N(0, \Sigma(v))$ where $\Sigma^{-1}(v)$ is the Fisher information matrix. Asymptotically $\hat{I}_\Delta(v)$ is an unbiased estimator of $I(v)$, i.e. , $\mathbf{E}\{\hat{I}_\Delta(v)\} = I(v)$.

3. DISCRETE STOCHASTIC OPTIMIZATION BASED HMM ALGORITHM

3.1. Discrete Stochastic Optimization Problem

Determining the Nernst potential v^* requires conducting experiments at different values of voltage v. In patch clamp experiments, the applied voltage v is chosen from a finite set. Let $v \in V = \{\theta(1), \ldots, \theta(M)\}$ denote the finite set of possible voltage values that the experimenter can pick. Determining $v^* \in V$ can be formulated as a discrete optimization problem: $v^* = \arg\min_{v \in V} |I(v)|^2$. Due to the presence of large amounts of thermal noise, $I(v)$ cannot be exactly evaluated and only unbiased estimates $\hat{I}(v)$ are available. Computing the Nernst potential is equivalent to the following discrete stochastic optimization problem:

$$\text{Compute } v^* = \arg\min_{v \in V} \left[\mathbf{E}\{\hat{I}(v)\}\right]^2 \qquad (3)$$

where $\hat{I}(v)$ is the MLE of the parameter $I(v)$ of the HMM. Since no closed form expression is available for the above expectation, we need to develop a simulation based (stochastic approximation) algorithm.

HMM Brute Force Approach: A brute force approach for solving (3) involves an exhaustive enumeration as follows:

For each $v \in V$, run an independent experiment to gather the sample path $\{y_1(v), y_2(v), \ldots, y_\Delta(v)\}$ for a very large batch size Δ. Compute the MLE $\hat{I}(v)$ via a HMM parameter estimator. Finally pick $\hat{v}^* = \arg\min_{v\in V} |\hat{I}(v)|^2$. Since for any fixed $v \in V$, the MLE $\hat{I}(v)$ is strongly consistent [6], $\hat{I}(v) \to I(v)$ w.p.1, as the batch size $\Delta \to \infty$. Thus

$$\arg\min_{v\in V}\big(\hat{I}(v)\big)^2 \to \arg\min_{v\in V}\big(I(v)\big)^2 \text{ w.p.1.} \qquad (4)$$

Thus in principle, the above brute force simulation method can solve the discrete stochastic optimization problem (3) for large Δ and the estimate is *consistent*, i.e. , (4) holds. However, it is highly inefficient since $\hat{I}(v)$ needs to be evaluated for each $v \in V$. The evaluations of $\hat{I}(v)$ for $v \neq v^*$ are wasted because they contribute nothing to the estimation of $i(v^*)$ at the Nernst potential v^*.

3.2. Discrete Stochastic Approximation Algorithm

We propose a discrete stochastic approximation algorithm that is both *consistent* (i.e. , (4) holds) and *attracted* to the Nernst potential. That is, the algorithm should spend more time gathering observations $\{y_n(v)\}$ at the Nernst potential $v = v^*$ and less time for other values of $v \in V$. Thus in discrete stochastic approximation the aim is to devise an *efficient* adaptive search (sampling plan) which allows to find the minimizer v^* with as few samples as possible by not making unnecessary observations at non-promising values of v.

Notation: Let $n = 1, 2, \ldots$ denote discrete time. Group the discrete time into batches of length Δ – typically $\Delta = 10,000$ in experiments. Let $N = 1, 2, \ldots$ denote batch number, batch N comprises of the Δ discrete time instants $n \in \{N\Delta, N\Delta + 1, \ldots, (N+1)\Delta - 1\}$. Let $D_N = (D_N(1), \ldots, D_N(M))'$ denote the vector of duration times the algorithm spends at the M possible potential values in V. Finally define the M dimensional unit vectors, e_m, $m = 1, \ldots, M$ as $e_m = \begin{bmatrix}0 & \cdots & 0 & 1 & 0 & \cdots & 0\end{bmatrix}'$ with 1 in the m-th position and zeros elsewhere.

The discrete stochastic approximation algorithm of [3] is not directly applicable to the cost function (3) – since it applies to optimization problems of the form $\min_{v\in V} \mathbf{E}\{C(v)\}$. However, (3) can easily be converted to this form as follows: Let $\hat{I}_1(v)$, $\hat{I}_2(v)$ be two statistically independent unbiased HMM estimates of $I(v)$. Then defining $\hat{C}(v) = \hat{I}_1(v)\hat{I}_2(v)$, it straightforwardly follows that $\mathbf{E}\{\hat{C}(v)\} = \left[\mathbf{E}\{\hat{I}(v)\}\right]^2 = |I(v)|^2$. We propose the following discrete stochastic approximation algorithm:

Algorithm 1 [Algorithm for Learning Nernst Potential]

`Step 0:` (Initialization.) At batch-time $N = 0$, select starting point $X_0 \in \{1, \ldots, M\}$ randomly. Set $D_0 = e_{X_0}$, Set initial solution estimate $\widehat{v}_0^* = \theta(X_0)$.

`Step 1:` (Sampling.) At batch-time N, sample $\tilde{X}_N \in \{X_N - 1, X_N + 1\}$ with uniform distribution.

`Step 2:` (Evaluation and Acceptance.) Apply voltage $\tilde{v} = \theta(\tilde{X}_N)$ to patch clamp experiment. Obtain two Δ length batches of HMM observations. Let $\hat{I}_N^{(1)}(\tilde{v})$ and $\hat{I}_N^{(2)}(\tilde{v})$ denote the HMM-MLE estimates for these two batches which are computed using the EM algorithm of Appendix A. Set $\hat{C}_N(\tilde{v})) = \hat{I}_N^{(1)}(\tilde{v})\hat{I}_N^{(2)}(\tilde{v})$.

Then apply voltage $v = \theta(X_N)$. Compute the HMM-MLE estimates for these two batches, denoted as $\hat{I}_N^{(1)}(v)$ and $\hat{I}_N^{(2)}(v)$. Set $\hat{C}_N(v)) = \hat{I}_N^{(1)}(v)\hat{I}_N^{(2)}(v)$.

If $\hat{C}_N(\tilde{v}) < \hat{C}_N(v)$, set $X_{N+1} = \tilde{X}_N$, else, set $X_{N+1} = X_N$.

`Step 3:` (Update occupation probabilities of X_N.) $D_{N+1} = D_N + e_{X_{N+1}}$

`Step 4:` (Update estimate of Nernst potential.) $\widehat{v}_N^* = \theta(m^*)$ where $m^* = \arg\max_{m\in\{1,\ldots,M\}} D_{N+1}(m)$, set $N \to N+1$, go to Step 1.

Above $\hat{v}_N^*$ denotes the estimate of the Nernst potential at batch N. We will show below that $\hat{v}_N^* \to v^*$ w.p.1, meaning that the algorithm is both attracted to the maximum.

3.3. Convergence and Attraction of Algorithm 1

Throughout this section we assume:

(N) The batch size Δ is sufficiently large so that $\hat{I}_N^{(1)}(v)$, $\hat{I}_N^{(2)}(v)$ are $N(I(v), \Sigma(v))$ Gaussian random variables.

(O) For $m \in \{1, \ldots, M-1\}$, $I^2(\theta(m+1)) > I^2(\theta(m))$ implies $P\Big(\hat{C}(\theta(m+1)) > \hat{C}(\theta(m))\Big) > 0.5$. $I^2(\theta(m+1)) < I^2(\theta(m))$ implies $P\Big(\hat{C}(\theta(m+1)) > \hat{C}(\theta(m))\Big) < 0.5$.

Theorem 1 *Under the condition (O) above, the sequence $\{\theta(X_N)\}$ generated by Algorithm 1 is a homogeneous, aperiodic, irreducible Markov chain with state space V. Furthermore, Algorithm 1 is attracted to the Nernst potential v^*, i.e. , for sufficiently large N, the sequence $\{\theta(X_N)\}$ spends more time at v^* than an other state.*

PROOF. Given that the objective function is exactly of the form of the cost function in [3], we only need to verify condition (O). It then follows from [3, Theorem 2.1] that Algorithm 1 converges to a local minimum. However, $I(v)$ is a monotonically increasing function of v. Hence $I^2(v)$ has only a single minimum which is the global minimum. Thus under condition (O), Algorithm 1 converges to the Nernst potential. To verify condition (O), recall that $\hat{C}(v) = \hat{I}_1(v)\hat{I}_2(v)$ where $\hat{I}_1(v), \hat{I}_2(v) \sim N(I(v), \Sigma(v))$ due to normality assumption (N). Expanding out $\hat{C}(v) = \hat{I}_1(v)\hat{I}_2(v)$ yields $\hat{C}(v) = I^2(v) + W(v)$ where $W(v)$ is a

zero mean random variable with symmetric probability density function. Condition (O) then holds as proved in Lemma 3.1 of [7] for symmetric probability density functions. □

4. NUMERICAL RESULTS

We simulated sample paths of the ion channel current $\{i_n(v)\}$ as a Markov chain with transition probability matrix A (see (1)) and open state current level $I(v)$. Here

$$A = \begin{bmatrix} 0.97 & 0.03 & 0 \\ 0.3 & 0.6 & 0.1 \\ 0 & 0.1 & 0.9 \end{bmatrix} \quad (5)$$

The observed channel current at the electrode was simulated by adding white Gaussian noise with $\sigma(v) = 0.3$ to the ion channel current sequence $\{i_n(v)\}$, resulting in the HMM sequence $\{y_n(v)\}$ (see (2)).

We used Algorithm 1 to determine the Nernst potential v^*. Experiments were run over batch sizes $\Delta = 10,000$. At Step 0, we selected the starting point at $X_0 = 1$, i.e. , initial applied voltage $v = -160$ mV. In Step 2, the EM algorithm was run for 500 iterations on each Δ-length batch of HMM observations. As described at the end of Sec.3.2, this takes only about 0.002 secs on a 2 GFlop Pentium 4. The resulting MLEs for the 4 batches, namely $\hat{I}_N^{(1)}(\tilde{v})$, $\hat{I}_N^{(2)}(\tilde{v})$, $\hat{I}_N^{(1)}(v)$ and $\hat{I}_N^{(2)}(v)$ were used to determine $\hat{C}_N(\tilde{v})$ and $\hat{C}_N(v)$.

Fig.1 shows the Nernst potential estimates $\widehat{v}_N^*$ generated by Algorithm 1 for batch-times $N = 0, 1 \ldots, 10000$. As can be seen, the estimate $\widehat{v}_N^*$ rapidly converges to the Nernst potential $v^* = -64$ mV.

To illustrate the attraction (learning) property of Algorithm 1, i.e. , it spends more time gathering information near the Nernst potential than other voltages, Fig.2 shows the occupation probabilities computed by Step 3. As shown in Fig.2, Algorithm 1 spends approximately 14% of its time at the Nernst potential. In comparison, a brute force HMM approach would spend equal resources at all voltages $v \in V$, i.e, its would spend $1/320$ of its time at the Nernst potential v^*. Thus Algorithm 1 is approximately 45 times more efficient than the brute force HMM approach. Equivalently, to get equally accurate estimates of the Nernst potential, the brute force HMM approach requires the patch clamp experiment to be run 45 times longer than the controlled patch clamp experiment that uses Algorithm 1.

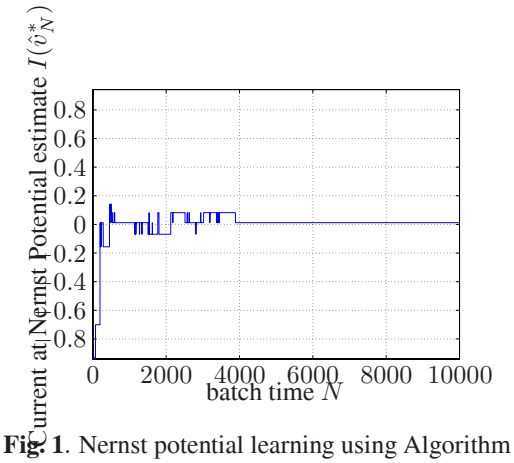

Fig. 1. Nernst potential learning using Algorithm 1.

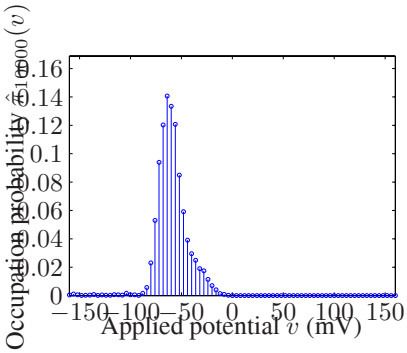

Fig. 2. Occupation probabilities of Algorithm 1.

5. REFERENCES

[1] E. Neher and B. Sakmann, "Single-channel currents recorded from membrane of denervated frog muscle fibres," *Nature*, vol. 260, pp. 799–802, 1976.

[2] S. Chung, V. Krishnamurthy, and J. Moore, "Adaptive processing techniques based on HMMs for characterising very small channel currents buried in noise and deterministic interferences," *Proc. Phil. Trans. Roy. Soc. Lond. B*, vol. 334, pp. 357–384, 1991.

[3] S. Andradottir, "A method for discrete stochastic optimization," *Management Science*, vol. 41, no. 12, pp. 1946–1961, 1995.

[4] ——, "Accelerating the convergence of random search methods for discrete stochastic optimization," *ACM Transactions on Modelling and Computer Simulation*, vol. 9, no. 4, pp. 349–380, Oct 1999.

[5] L. Venkataramanan, R. Kuc, and F. Sigworth, "Identification of hidden Markov models for ion channel currents – part ii: Bandlimited, sampled data." *IEEE Trans. Signal Proc.*, vol. 48, no. 2, pp. 376–385, Feb. 2000.

[6] P. Bickel, Y. Ritov, and T. Rydén, "Asymptotic normality of the maximum-likelihood estimator for general hidden Markov models," *Annals of Statistics*, vol. 26, pp. 1614–1635, 1998.

[7] W. Gong, Y. Ho, and W. Zhai, "Stochastic comparison algorithm for discrete optimization with estimation," *SIAM Journal on Optimization*, vol. 10, no. 2, pp. 384–404, 1999.