# Statistical Case Study on MLE for Stochastic Resonate-and-Fire Neuronal Model

Xandre Clementsmith

April 29, 2021

## 1 Summary

### 1.1 Theoretical Introduction

In the field of computational neuroscience, neuronal models are used to simulate the mechanisms of biological processes. The Hodgkin-Huxley model was one of the first of these models, based on the neurophysiological work done by Hodgkin and Huxley on the giant squid axon. However, parameter estimation and simulation is computationally expensive for this artificial neuron, preventing its use in creating large and complex networks. As a result, spiking models like integrate or resonate-and-fire neurons were developed. A previous study developed a quadratic model that is capable of reproducing almost all types of firing patterns while greatly increasing computational efficiency. For this work, the model is reduced to subthreshold activity, allowing for parameters that are voltage-independent. In addition, it becomes possible to simulate model dynamics with linear equations.

Several approaches have been used for parameter estimation from voltage trace. For instance, integrate-and-fire neurons have been hand-tuned to Hodgkin-Huxley models, and entire exploration of the parameter space has been used for single-compartment model neurons. These are either labor intensive methods, or become increasingly difficult the higher the dimension. Maximum Likelihood Estimation (MLE) presents an alternative to these approaches. MLE has previously been used for stochastic integrate-and-fire models. For this study, MLE is used to optimize the parameters of a stochastic resonate-and-fire model with Gaussian distribution. The input and voltage traces are assumed to be known.

### 1.2 Methods

**Resonate-and-Fire Model.** A resonate-and-fire model is derived from a two dimensional quadratic model that accounts for potassium and sodium ion currents, as well as their inactivation. The model is as follows

$$\frac{dv}{dt} = k_1 v + k_2 - k_3 u + k_3 i \tag{1}$$

$$\frac{du}{dt} = a(bu - v) \tag{2}$$

where $i$ is the input and $v$ is the potential. After a spike is fired, u and v are reset, such that $v \to c$ and $u \to u + d$. Spiking is represented as a binary, and post-threshold dynamics are unaccounted for otherwise. The threshold is dependent on the instant voltage value, as well as the voltage rate of change before every action potential. There is no clear static threshold value, so it is assumed to be *i.i.d.* distributed on a Gaussian. Simulated annealing for MLE was utilized in tandem with least-squares to estimate parameter values at each iteration.

**Parameter Separation and Least Squares.** There are 9 parameters that require estimation: $k1, k2, k3, a, b, c, d, m, \sigma$. Data is required for this sort of optimization. In particular, interspike intervals (ISIs) and membrane potential depolarization recordings are utilized. The parameters are separated into a set described by linear dynamics $\theta_l = (k1, k2, k3, a, b)$ and a set linked to after-spike resetting and threshold variation $\theta_t = (c, d, m, \sigma)$.

Linear systems can be analytically solved by methods involving gradients or least squares. The linear system of $\frac{dv}{dt} = k_1 v + k_2 - k_3 u + k_3 i$, $\frac{du}{dt} = a(bu - v)$ can be solved by

$$v = f(i, t; \theta_l) \tag{3}$$

where $f$ is a nonlinear function. The values of $c$ and $d$ are ignored by choosing data beyond the immediate transient period following post-threshold dynamics, so that the theta sets are independent. In this case, $\theta_l$ is solved using least squares, run iteratively. An objective function, otherwise known as a loss function, is minimized to determine an optimal solution. In this case, the function is

$$S = \sum_i (v_i - \hat{v}_i)^2 \tag{4}$$

This is simply a representation of least squares between prediction and actual. As this is an iterative algorithm, theta values are updated at every step. This is modeled by this series of equations.

$$\theta_l^{k+1} = \theta_l^k + \Delta\theta_l \tag{5}$$

$$\Delta\theta_l = (J'J)^{-1} J' \Delta v \tag{6}$$

$$\Delta v = v - \hat{v} \tag{7}$$

$$J_{ik} = \frac{\delta \hat{v}_i}{\delta \theta_{lk}} \tag{8}$$

Here, $J$ is a Jacobian matrix, and $\Delta v$ defines the error between the actual and estimate.

**Maximum Likelihood Estimation**

For MLE, $\theta_t$ is used, where $\{t_j\}$ represents the peak of each spike, and the interspike time ($t \in (t_{j-1}, t_j)$) is defined as $\{t_i\}$. The likelihood is written as

$$L(\theta_t) = log(\Pi_i \int_{vi}^{\infty} G(V_t; m, \sigma) dV_t * \Pi_j \int_{-\infty}^{v_j} G(V_t; m, \sigma) dV_t) \tag{9}$$

$$= \Sigma_i log \int_{v_i}^{\infty} G(V_t; m, \sigma) dV_t + \Sigma_j log \int_{-\infty}^{v_j} G(V_t; m, \sigma) dV_t \tag{10}$$

such that the likelihood is defined as the probability that $V_t$ is under membrane potential $v$ at $\{t_j\}$ and is over $v$ at $\{t_i\}$. $G(V_t; m, \sigma)$ is defined as the Gaussian pdf of $V_t$ with mean $m$ and variance $\sigma^2$.

**Simulated Annealing**

Simulated annealing (SA) is used for optimization in the case of MLE. It is notated as

$$P = \begin{cases} 1 & L(s') > L(s) \\ e^{-1/T} & otherwise \end{cases} \tag{11}$$

where a point s' in the neighborhood of s is randomly selected. T is a decreasing variable based on the concept of a cooling "temperature" in the case of metallurgic annealing.

$$T = T_0(1 - r)^2 \tag{12}$$

The variable $r$ represents the ratio of current iterations $n$ to the maximum number of iterations $N$. The temperature and random selection of $s'$ occur iteratively.

### 1.3 Results

The parameter estimation is performed on both the resonate-and-fire model, and the Hodgkin-Huxley model. Data is generated in the NEURON environment, with a current clamp involving white noise injection as input. Data is sampled at 50kHz. A comparison is made to existing generated data, allowing for a benchmark for the method. The prediction and actual data are close to identical, indicating the robustness of the method. The error between the two time series is under 5%. This is the case for both the resonate-and-fire method, as well as the Hodgkin-Huxley method. Data for linear modeling was only selected between spikes, with a great deal of data directly before and after action potentials. These piecewise traces are important for parameter separation. No other results are provided besides the performance on novel data.

## 2 Critique

### 2.1 Data Analysis

Membrane potential depolarization recordings are used for analysis, as interspike intervals are viewed to require more complicated estimation methods. This immediately creates a small problem, as the reason for developing these models is to model a variety of spiking activities without using a complex neuron model. However, it is likely that the data are still reproducible for spiking variants, and it is only the choice of data source that is restricted. The program NEURON was used for creating comparative data sets. In particular, it was used to simulate typical Hodgkin-Huxley and Resonate-and-Fire neurons, to compare their results to parameter estimation by MLE and least squares. As this is a computational

improvement rather than focusing on actual neurophysiology, this is a sensible choice. It's particularly impressive that this pair of equations can match the Hodgkin and Huxley model so accurately.

## 2.2 Methods

**Parameter Separation and Least Squares.** As stated, the parameters are separated into two sets. This separation of parameters allows for a great deal of efficiency, as the set of linearly solvable values are computed using least squares, rather than MLE. Additionally, the choice to exclude the transient is commonplace in computational neuroscience. Not only does it allow for computational efficiency by linearly separating the parameters, it creates a more homogeneous dataset for estimation.

The choice for using linear squares to solve linear dynamics is fairly standard. Given that the distribution is normal, linear squares is comparable to MLE. At each iteration, solutions are simply additive, which is to be expected. The change in $\theta$ is slightly more complex, utilizing jacobian matrices, as well as their transposes and inverses. This equation is obtained by setting the gradient of the loss to zero and solving in the case of linear least squares. Likely the most computationally expensive component would be the calculation of gradients for the Jacobian matrices, which would be determined once per iteration for each neuron.

One interesting decision was the choice of using the Guassian distribution, which I don't believe is justified anywhere. Certainly, Gaussian is a common choice, and tends to be the default on various programs and toolboxes. However, it is known that spike trains tend to follow a Poisson process. Was this distribution considered? This would seem like an especially appealing choice given that the data is discrete across time. Even if there is a compelling argument against the use of Poisson, there should be a small section dedicated to explaining the choice of distribution, or a comparison between approaches for this case.

**Maximum Likelihood Estimation**

Using the log-likelihood in this case is a good choice, due to it's computational convenience over a non-logarithmic based likelihood function. As the Gaussian distribution is a part of the exponential family of distributions, it is logarithmically concave, which is helpful for MLE. I would suggest that they denote log likelihood with a lower-case $l$, as that is the convention I have seen for it to differentiate it from its counterpart. However, the paper does indicate that they are using the term likelihood as a simplification of log-likelihood. Perhaps they are just doing the same thing with notation. For this calculation, $\sigma_l$ is already computed and considered to be fixed, such that a new voltage trace needs to be generated. The final notation for MLE of $\theta_t$ is given: $\hat{\theta}_t = \arg\max L(\theta_t)$. This is written without an included variable for the data sample. Again, this is more a problem of what I understand to be standard notation for log-likelihood, rather than the actual methodology. Furthermore, the use of Gaussian is assumed, rather than justified.

The authors note that computation time is mostly composed of calculating the likelihood function. Of course, any sort of algorithm that works non-linearly is expected to have a large time complexity than a linear method. As far as I can tell, no major efforts are made to reduce this. One obvious idea is to make the maximum number of iterations for simulated annealing a function rather than a flat value, so as to prevent redundant calculations. As this paper is

merely an improvement on an existing set of functions in computational neuroscience, that level of mathematical creativity or hacking isn't to be expected.

**Simulated Annealing**

As MLE cannot be solved explicitly, iterative optimization algorithms like gradient descent are used for maximizing the likelihood. In this case, simulated annealing (SA) is utilized due to its ability to escape local maxima in search of greater global values. In particular, methods like gradient descent excel at finding precise local extremities in a fixed amount of time, while SA will merely approximate a global optimum. I think it's choice is understandable for research purposes, given the lack of time restraints and the importance of maximizing the likelihood. However, it might be favorable to use something like gradient descent for very large networks. I think that the optimal way to determine the best optimization algorithm for this process is by standardized benchmarking, like in the case of the deep learning community. Hundreds of very specific variations have been made to common algorithms to improve efficiency and accuracy thanks to common benchmarks and a large community of developers. Such rapid developments are unnecessary in this case, however more justification for SA from the authors would be preferred. The equations aren't representative of actual neuron dynamics like in the case of Hodgkin-Huxley neurons, and instead are simply meant to produce similar results. Are global maxima more or less important with this in consideration? The paper gives an explanation of the SA algorithm, though the description and mathematics are identical to previous literature and require little review here. This is especially true as variables such as "temperature" are in this case entirely theoretical, and have no relation to the actual data or subject matter.

### 2.3 Results

For a scientific paper or from a data science perspective, the results were rather minimal. They did cover the main hypothesis, and showed a great deal of predictive accuracy for the novel technique. This was accomplished for both neural models. Still, more results would be appreciated, especially regarding comparisons. For instance, how do other statistical models or optimization algorithms compare to those used? Furthermore, a paragraph discussing the time and space complexity and a comparative analysis to previous methods would be beneficial. Overall however, I think that the use of MLE for this task was warranted and produced meaningful results.

## 3 Further Research

One major area for further study would be a comparative approach for optimization, and other parameter estimation techniques. For instance, how does SA compare to gradient descent? Is MLE the best approach for estimating the non-linear parameters, or might something like Bayesian parameter estimation work well? This could involve the metrics used in this study, specifically the predictive capabilities of the model. Preferably, this would be coupled with a time and space complexity estimate. The reason for this is that certain statistical methods work well on particular types of data, but vary in their performance when introduced to other data sets. It's hard to determine which methods are most

efficient and capable without an actual empirical comparison. Furthermore, an analysis of time complexity would indicate the improved computation capabilities of using MLE for parameter estimation in this resonate-and-fire model.

Another obvious continuation of this work would be to study its capabilities in other neuron dynamics, and other data types. From what I could tell, spiking and subthreshold dynamics were analyzed and predicted for one type of neuron output, whereas there are a variety of possible neural behaviors. For instance, how well does MLE estimate the parameters of bursting neurons, that release a large amount of action potentials in a short interval? These variations could be analyzed with novel data sets as well, like utilizing interspike intervals, or even entirely different samplings like calcium fluorescence. Strong performance on a diversity of models and data types would indicate the robustness of MLE and SA for simulating neural dynamics efficiently.

One particular difficulty in understanding neural dynamics is the task of identifying causal couplings of neurons, where one neuron is synaptically connected to another. This would be indicated by direct information flow from one neuron to another. The issue is that this becomes increasingly difficult as the size of the network increases, as the multivariate system can be subject to a great deal of multicollinearity. Perhaps using MLE in a simulation of a network would help provide an effective or causal graph of the neural circuit, as a greater degree of parameters and detail would be present for analysis. At the very least, MLE can be used to help create more accurate and efficient simulations where other methods like Transfer Entropy can be utilized for inferring network dynamics.

The field of computational neuroscience benefits substantially from these marginal improvements in modeling complex biological systems. This study was an attempt to optimize previous work, which provided a method for using a quadratic system to mimic a much more complex neuron model by Hodgkin and Huxley. While there are some approaches that can expand on this algorithm, it is likely that future work will instead utilize it for related but ultimately divergent fields in computational neuroscience. For instance, it may be written into the source code for NEURON, allowing anyone to simulate a network of complex Hodgkin-Huxley neurons with a few spare minutes and a laptop. Indeed, a majority of models and methods in computational neuroscience rely heavily on work in statistics. This work presents a strong application of MLE for an important subset of neuroscientific study.

# Maximum Likelihood Parameter Estimation in a Stochastic Resonate-and-Fire Neuronal Model

Jun Chen*, Jose Suarez†, Peter Molnar‡ and Aman Behal§
*Department of Electrical Engineering and Computer Science
University of Central Florida, Orlando, Florida 32817
Email: lay2008@knights.ucf.edu
†Department of Electrical Engineering and Computer Science
University of Central Florida, Orlando, Florida 32817
Email: jnsuarez@knights.ucf.edu
‡Department of Zoology, Division of Natural Sciences
University of West Hungary, Savaria Campus, Karolyi Gaspar ter 4. Szombathely, H-9700, Hungary
Email: pmolnar@pminfonet.com
§Department of EECS and NanoScience Technology Center
University of Central Florida, Orlando, Florida 32817
Email: abehal@mail.ucf.edu

***Abstract*—Recent work has shown that resonate-and-fire model is both computationally efficient and suitable for large network simulations. In this paper, we examine the estimation problem of a resonate-and-fire model with random threshold. The model parameters are divided into two sets. The first set is associated with subthreshold behavior and can be optimized by a nonlinear least squares algorithm. The other set contains threshold and reset parameters and its estimation is formulated in terms of maximum likelihood formulation. We evaluate such a formulation with detailed Hodgkin-Huxley model data.**

***Keywords*-resonate-and-fire; parameter estimation; maximum likelihood; simulated annealing.**

## I. Introduction

One essential issue in computational neuroscience is to characterize the relationship between neural output recording and the input current to the cell, [1]. To reproduce the behavior of neurons, the spiking neuron model has been widely used. Although a detailed Hodgkin-Huxley [2] neuron model could mimic the neuronal dynamics more accurately, it is computationally inefficient in parameter estimation and model simulation. Therefore, spiking model is the first choice for large network simulation. In [3], a second order model is introduced to be able to reproduce almost all types of firing patterns *in-vivo* and keep the computation efficient at the same time. In authors' previous work, a linear-in-the-parameters presentation of aforementioned quadratic model is developed and allows us to identify experimentally obtained data. However, a small flaw in the assumptions prevents us from further research. Although the quadratic model is proven to be biologically meaningful, it can only qualitatively reproduce the firing pattern, i.e., the model can not quantitatively represent the upstroke of the spike unless it is assumed that the parameters are voltage-dependent [4]. To address this issue, a modification is made to this model. Instead of assuming voltage-dependent parameters, we remove the quadratic term in the model and consider it to be a representation only for subthreshold dynamics. Since now we narrow our interest zone down to the subthreshold region, the mismatch in the upstroke does not affect our identification any more. Another reason to consider simplifying the model instead of additional assumption is, by the removal, the model dynamics become linear and can be treated analytically, without hurting the ability to replicate multiple firing patterns.

In order to match the experimentally observed data to a particular parametric model , many parameter estimation methods have been introduced for estimating parameter values from the recorded voltage trace. In [5], an adaptive exponential integrate-and-fire neuron model is manually hand-tuned to fit a detailed Hodgkin-Huxley based model. Although the result in [5] shows good match between the proposed model and the detailed Hodgkin-Huxley based model, such trial-and-error approach depends mainly on researcher's experience and thus is labor-intensive. In [6], a database of single-compartment model neurons is constructed by exploring the entire parameter space – this approach is only practical when the parameter space has a low dimension. In [1], a maximum likelihood problem is formulated with a stochastic integrate-and-fire model. The maximum likelihood formulation can be also found in other literature, e.g., [7].

In this paper, we consider a resonate-and-fire model with stochastic threshold which is assumed to be a Gaussian random variable. By assuming that the threshold is the only component that has a stochastic property, the membrane potential is therefore deterministic and solvable. Here, we

assume that both the input to the cell and the voltage recording trace are known for characterization. A nonlinear least squares method is first applied to fit the subthreshold dynamics. Then, the second characterization problem consists of estimating of parameters that associate with the initiation of the firing event and the reset. Considering the interspike interval as only reference in this level of estimation, we optimize the parameters by finding the maximum of the likelihood of the obtained firing data through simulated annealing. We apply this formulation on detailed Hodgkin-Huxley model data and show preliminary results.

The remainder of this paper is organized as follows. In Section II, we present the resonate-and-fire neuron model with random threshold. Technical details for the identification mechanism are provided in Section III. Section IV provides preliminary results followed by discussion in Section V. Appropriate conclusions are drawn in Section VI.

## II. Problem Definition

A two dimensional quadratic neuron model is introduced in [3]. Compared to the integrate-and-fire model, the additional variable, accounting for the activation of $K^+$ ionic currents and inactivation of $Na^+$ ionic currents, allows the model to present many types of firing patterns that are common in biological systems. However, the nonlinearity introduced by the quadratic term prevents one from treating the model analytically. Meanwhile, it also requires variable parameters to reproduce the exact spike shape [4]. Therefore, we remove the quadratic term and make the model as resonate-and-fire type, as follows

$$\frac{dv}{dt} = k_1 v + k_2 - k_3 u + k_3 i \tag{1}$$

$$\frac{du}{dt} = a(bu - v) \tag{2}$$

when $i$ is the input/synaptic current. Whenever the potential $v$ hits the threshold, the neuron is said to fire a spike, and the variables are reset according to

$$if\ v = V_t,\ then \begin{cases} v \to c \\ u \to u + d \end{cases} \tag{3}$$

where $V_t$ is the threshold. Although it is in a different formulation, the model present here is the same as the one discussed in [8] or [9].

Note that (1) and (2) only describe the subthreshold approximation, and the post threshold dynamics of cell does not form part of the model: the upstroke/downstroke are drawn by hand. Unlike [10] in which a triangular pulse is used to mimic the shape of the spike, here we ignore the whole spiking behavior, only using a straight line to mark the spike arrival.

Therefore, a threshold is needed to indicate the initiation of the spike. In [11], a fixed threshold is exploited for this purpose. However, the voltage threshold for a spike depends not only on the instant voltage value, but also on the rate of voltage change preceding each spike, which indicates that a variable threshold is more realistic. The variation of spike threshold could also be a function of instantaneous firing rate [12]. Fig. 1 shows the probability of firing for specific voltage value, from which we can see there is no obvious "jump" in firing probability so that we can define it as "threshold". Data used to generate Fig. 1 is obtained from a detailed ion-channel based spiking model of [13]. Actually, the stochastic property of spike threshold has been proposed and proved in literature [14], [15]. This randomness could be a consequence of signal noise or the aforementioned factors affecting the spike threshold. Others may use a stochastic term in membrane dynamics to present the signal noise, as in [7]. In fact, this is computationally identical to subtracting the noise from the potential and adding it to the threshold [16]. In this paper, we consider the threshold to be the random component in the neuron model and assume its distribution is a Gaussian and i.i.d. That is,

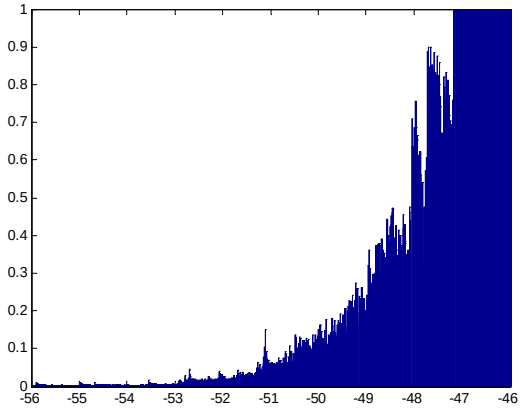


Figure 1. Firing probability of voltage shows there is no clear value could be defined as "threshold". For each voltage value, if there are $N$ points at this value in the database and $n$ of them directly followed by a spike, the the firing probability is defined as $n/N$.

$$V_t \sim N(m, \sigma) \tag{4}$$

## III. The Estimation Problem

Our problem now is to estimate the parameters $(k_1, k_2, k_3, a, b, c, d, m, \sigma)$. Generally, two types of data can be used for the identification, interspike intervals (ISIs) and membrane potential depolarization recording. For the first type, the estimation problem would be more complicated and need additional assumptions to succeed [7]. Therefore, we investigate the second kind of data here. We divide the parameters into two sets: $\theta_l = (k_1, k_2, k_3, a, b)$ is associated with linear dynamics while $\theta_t = (c, d, m, \sigma)$ is linked to after-spike resetting and threshold variation. Thus, the

estimation problem consists of both matching the voltage recording to find $\theta_l$ and maximizing the likelihood to find $\theta_t$.

### A. Subthreshold Estimation

Linear system given by (1) and (2) can be analytically treated and solved. This is useful when we want to use gradient based or least squares based estimation technique, which requires the derivative of the objective function with respect to the parameters. The solution of (1) and (2) can be presented as

$$v = f\left(i, t; \theta_l\right). \tag{5}$$

Given the input $i$, $f$ is a nonlinear function on parameter set $\theta_l$. In fact, $v$ also depends on $c$ and $d$ which we do not identify in this stage. The effect of $c$ and $d$ will be excluded by choosing data beyond the transient period immediately following a spike. Thus in the subthreshold region, the voltage dynamic is independent of $\theta_t$, depending only on $\theta_l$. The estimate for $\theta_l$ is therefore given by nonlinear least squares estimation [17]. Non-linear least squares is a form of least squares used to fit observations to a model that is nonlinear in parameters. The algorithm is run iteratively. If the reference data is a $n \times 1$ vector, then the objective function to be minimized is

$$S = \sum_i \left(v_i - \hat{v}_i\right)^2 \tag{6}$$

and the update law is given by

$$\theta_l^{k+1} = \theta_l^k + \Delta\theta_l \tag{7}$$

$$\Delta\theta_l = \left(J'J\right)^{-1} J'\Delta v \tag{8}$$

where $\Delta v = v - \hat{v}$ is a $n \times 1$ vector defining the error between actual and estimate; and $J$ is the $n \times j$ Jacobian matrix, with elements given by

$$J_{ik} = \frac{\partial \hat{v}_i}{\partial \theta_{l_k}}. \tag{9}$$

Note that Jacobian matrix $J$ is a function of estimated parameters and updates from one iteration to the next.

### B. Maximum Likelihood Estimation

In this step, we assume the interspike intervals are the only output recording from the cell. The spike time $\{t_j\}$ is defined to be the time of the peak of each spike. During interval $t \in (t_{j-1}, t_j)$, the cell is not firing. Since the data we examine here are discrete-time sample data, we denote all data points during interval $(t_{j-1}, t_j)$ as $\{t_i\}$. The likelihood for parameter $\theta_t$ can therefore be defined as the probability that the noisy threshold $V_t$ is under the membrane potential $v$ at spike time $\{t_j\}$ and over $v$ at non-spike time $\{t_i\}$, which can be written as

$$\begin{aligned} L\left(\theta_t\right) &= \log\left(\prod_i \int_{v_i}^{\infty} G\left(V_t; m, \sigma\right) dV_t \right. \\ &\qquad \left. * \prod_j \int_{-\infty}^{v_j} G\left(V_t; m, \sigma\right) dV_t \right) \\ &= \sum_i \log \int_{v_i}^{\infty} G\left(V_t; m, \sigma\right) dV_t \\ &\quad + \sum_j \log \int_{-\infty}^{v_j} G\left(V_t; m, \sigma\right) dV_t \end{aligned} \tag{10}$$

where $G\left(V_t; m, \sigma\right)$ is the Gaussian probability density function of threshold $V_t$ with mean $m$ and variance $\sigma^2$. Technically, the term "log-likelihood" would be more appropriate to denote $L\left(\theta_t\right)$, but in this paper we use "likelihood" for simplicity. Note that since $v$ is a function of $c$ and $d$, since $\theta_l$ is considered fixed at this stage, we still need to generate a new voltage trace for each set of parameter $\theta_t$.

Based on the likelihood function (10), the estimated parameter set is thus given by

$$\hat{\theta}_t = \arg\max L\left(\theta_t\right). \tag{11}$$

### C. Simulated Annealing

In this paper we use simulated annealing (SA) technique [18], [19] to find the maximum of the previously formulated likelihood. Unlike gradient based law or least squares which might be stuck with local maxima, simulated annealing is designed for global optimization. The name "simulated annealing" comes from the annealing technique in metallurgy, which involves heating and controlled cooling of a material to minimize the internal energy of its crystals. Fig. 2 shows the SA algorithm. In each iteration, a point $s'$ in the neighborhood of the old point $s$ is randomly selected. The decision to consider or reject the new point as a "better" point depends on the probability function $P$

$$P = \begin{cases} 1 & \text{if } L\left(s'\right) > L\left(s\right) \\ e^{-1/T} & otherwise \end{cases} \tag{12}$$

where $T$ is called "temperature." The possibility to accept the worse point allows the algorithm to move along the opposite direct and thus get rid of local maximum.

The SA algorithm begins with a large temperature value $T_0$, allowing the algorithm to move randomly in the parameter space. Then the parameter gradually decreases. The "cooling" algorithm for the temperature $T$ used here is given by

$$T = T_0\left(1 - r\right)^2 \tag{13}$$

where $r$ is the ratio of current iteration $n$ to the maximum allowed iteration $N$. The algorithm stops after the maximum iteration is reached. A detailed description of the algorithm follows.

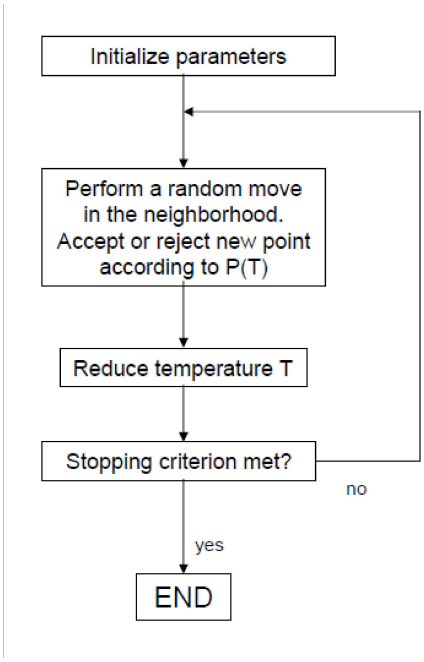


Figure 2. The simulated annealing algorithm

- *Step 0*. Choose a starting point $s_0$, a starting temperature $T_0$, and maximum iteration $N$. Set boundary for each direction if needed. Compute the likelihood of the initial point.
- *Step 1*. Generate a new point $s'$ from the neighborhood of point $s$. Projection applies if the new point exceeds the preset boundary.
- *Step 2*. Compute the likelihood of the new point. Call probability function (12), and replace $s$ with $s'$ if $P$ is larger than a randomly picked up number between 0 and 1. Otherwise, $s$ remains untouched.
- *Step 3*. Increase the current iteration $n$ and update the temperature $T$.
- *Step 4*. If $n < N$, go to step 1; else, stop the search.

## IV. Preliminary Results

We applied the aforementioned approach on two types of data, resonate-and-fire model data and detailed Hodgkin-Huxley model data. The detailed model data is generated in *NEURON* environment [20], by using a detailed ion-channel based spiking model of [13] with parameters from [21] (code for this model available at [22]). The model could simulate three types of spiking cell. In this paper we investigate the regular spiking cell. Use the current clamp simulation and inject pre-generated white noise current to stimulate the cell. Output data are sampled at 50kHz.

### A. Resonate-and-fire Model Data

We first apply proposed method on reference data generated by the exact model - this is to testify the correctness

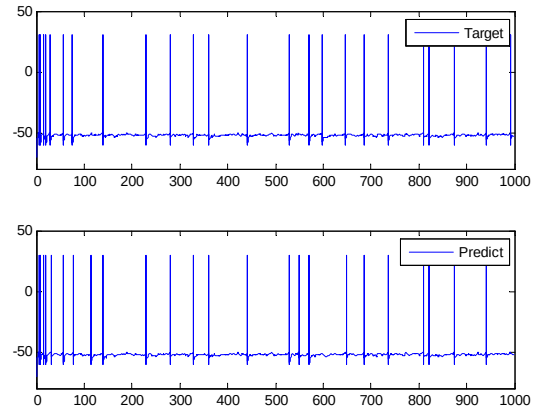


Figure 3. Comparison of spike train. Target data is generated by the exact model.

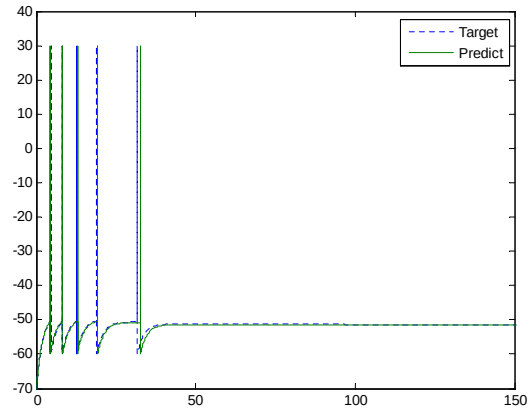


Figure 4. Comparison of spike train. Target data is generated with step input.

of the proposed formulation. Parameters for the reference model are chosen to simulate a rapidly adapting receptor (i.e., receptor initially responds and then quickly stops firing in response to steady input). Fig. 3 shows the evaluation result under noisy input. In Fig. 4, the estimated parameters successfully replicate the quick adapting firing pattern.

### B. Detailed Hodgkin-Huxley Model Data

Now,we present results on detailed Hodgkin-Huxley model data. After the estimation process, the estimated parameters are evaluated by comparing the predicted output of the model to unseen data. Fig. 5–6 show the evaluation results. From fig 5 one can see that the subthreshold traces are very close to each other, showing that dynamics (1) and (2) yield good approximation in the linear zone. Fig 6 shows part of the spike trains, in which the predicted train closely

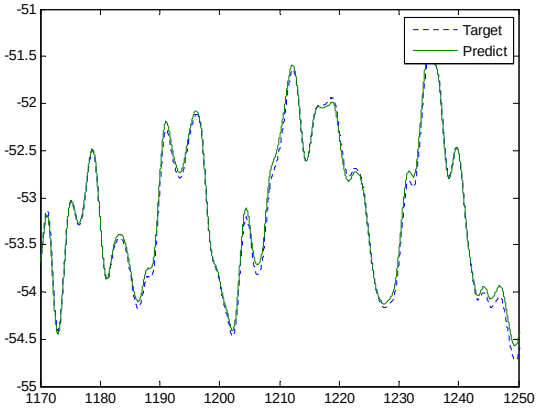


Figure 5. Comparison of subthreshold dynamics. Target data is generated under the same type of input as the one used in the reference data generation.

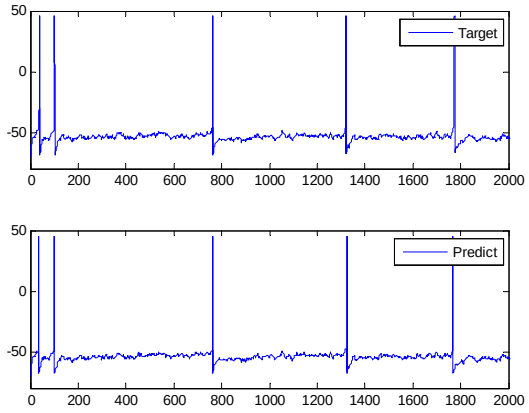


Figure 6. Comparison of spike train. Target data is generated under the same type of input as the one used in the reference data generation.

follows the target with the same adaptation rate. We ran the evaluation ten times, the statistics are listed in Table I. The estimated parameters lead to a averagely 16.3 spikes while the target has 15 spikes under the input stimulus.

Table I. Evaluation on HH Model

| No. of evaluations | Target spikes | Mean error |
|---|---|---|
| 10 | 15 | 1.3 |

Fig. 7 shows the evaluation result under step current. The prediction nicely follows the first three spikes, but misses the fourth one. However, the prediction still successfully replicates the spike adaptation encoded in the target train.

## V. Discussion

In our formulation, we assume that the subthreshold trace could be defined as a "linear zone" where the dynamics (1) and (2) are a good approximation. This subthreshold area is upperbounded by the variable threshold. Results shown in last section indicate the validity of this assumption, since the error between the predicted and target in Fig. 5 is under 5%. Since the linear model is capable of replicating only the subthreshold dynamics, the reference voltage trace used for the first-step nonlinear least squares estimation should not contain any supra-threshold activity. This is done by choosing piecewise traces between each spikes, with starting point and ending points far away from the nearest spike.

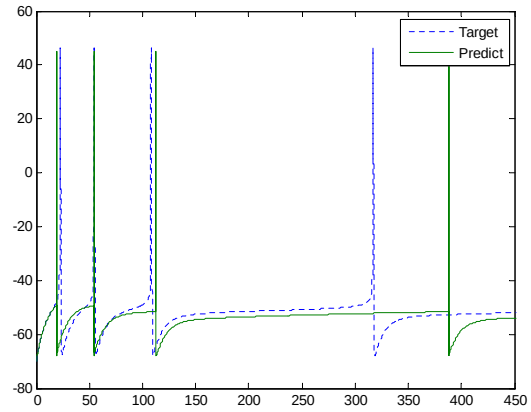


Figure 7. Predicted and target are under constant current input.

Another assumption we made in this formulation is that we assumed all the stochastic components in the system, including system noise and the variability of threshold, are presented by the threshold distribution. Our results suggest that such simplification leads to over prediction of the firing rate. However, the results are still acceptable - the errors are small in percentage and the predicted spike trains have similar patterns as the target. We believe that a model with more sophisticated assumption or component could address the over prediction issue, yet this would result in larger computation cost.

The computation of the likelihood function (10) is the major cost of computation. The original likelihood function is defined over all the time instants. The reference data is sampled at 50kHz, which results in a large size of $\{t_i\}$. Actually, considering the fact that $G(V_t; m, \sigma)$ is the Gaussian probability density function which is zero beyond $3\sigma$ distance from the mean $m$, thus the voltage trace was processed to make $\{t_i\}$ only contains points that are not too far away from the mean. This requires us to have some knowledge about the parameters before the estimation, which is acceptable since by looking at the reference train, one can easily draw a reasonable boundary for $m$ and $\sigma$.

## VI. Conclusion

In this paper, we studied a resonate-and-fire model with stochastic threshold. Considering the fact that a quadratic neuronal model is not capable of replicating the shape of spike, we removed the quadratic term from this model and made the model resonate-and-fire. Also considering the fact that channel noise exists in almost all cell systems, we assumed the threshold for the resonate-and-fire model to be random and normal. By formulating the parameter estimation problem into a two-stage problem, an estimation mechanism based on nonlinear least squares and maximum likelihood was presented. Reference data generated by a detail Hodgkin-Huxley model was fed to the estimation mechanism. Estimated parameters were evaluated by data that are not used in the estimation process. Results show good match between the prediction and the target.

## References


[1] L. Paninski, J. W. Pillow, E. P. Simoncelli, "Maximum Likelihood Estimation of a stochastic Integrate-and-Fire Neural Encoding Model, " Neural Computation, vol. 16, pp. 2533-2561, 2004.

[2] A. Hodgkin, A. Huxley, "A Quantitative Description of Membrane Current and Its Application to Conduction and Excitation in Nerve, " *J. Physiol*, vol. 117, pp. 500-544, 1952.

[3] E. M. Izhikevich, "Simple Model of Spiking Neurons," *IEEE Trans. Neural networks,* vol. 14, pp. 1569-1572, Nov. 2003.

[4] E. M. Izhikevich, *Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting*, The MIT Press, 2007.

[5] R. Brette, W. Gerstner, "Adaptive Exponential Integrate-and-Fire Model as an Effective Description of Neuronal Activity," *J. Neurophysiol* vol. 94, pp. 3637-3642, Jul. 2005.

[6] A. A. Prinz, C. P. Billimoria, E. Marder, "Alternative to Hand-Tuning Conductance-Based Models: Construction and Analysis of Database of Model Neurons," *J. Neurophysiol* vol.90, pp. 3998-4015, Aug. 2003.

[7] P. Lansky, P. Sanda, J. He, "The Parameters of the Stochastic Leaky Integrate-and-Fire Neuronal Model," *J Comput Neurosci*, vol. 21, pp. 211-223, 2006.

[8] G. Young, "Note on Excitation Theory, " *Psychometrika*, vol. 2, no. 2, 1937.

[9] R. FitzHugh, "Mathematical Models of Excitation and Propagation in Nerve, " In H. P. Schwan, *Biological Engineering*. New York: McGraw-Hill.

[10] R. Jolivet, T. J. Lewis, W. Gerstner, "Generalized Integrate-and-Fire Models of Neuronal Activity Approximate Spike Trains of a Detailed Model to a High Degree of Accuracy," *J. Neurophysiol* vol. 92, pp. 959-976, 2004.

[11] Eugene M. Izhikevich, "Resonate-and-fire Neurons," *Neural Networks*, vol. 14, pp 883-894, 2001.

[12] R. Azouz, C. M. Gray., "Dynamic Spike Threshold Reveals a Mechanism for Synaptic Coincidence Detection in Cortical Neurons *in vivo*, " *Proc. Natl. Acad. Sci. USA*, vol. 97, pp. 8110-8115, 2000.

[13] D. A. McCormick, Z. Wang, J. Huguenard, "Neurotransmitter Control of Neocortical Neuronal Activity and Excitability," *Cereb Cortex*. vol. 3, pp. 387-398. 1993.

[14] A. Bershadskii, E. Dremencov, G. Yadid, "Short-term Memory and Critical Clusterization in Brain Neurons Spike Series, " *Physics Letter A*, vol. 313, pp. 158-161, 2003

[15] John A. White, Ruby Klink, Angel Alonso, Alan R. Kay, "Noise From Voltage-Gated Ion Channels May Influence Neuronal Dynamics in the Entorhinal Cortex ," *J Neurophysiol*, vol. 80, pp. 262-269, 1998.

[16] Ian C. Bruce, Laurence S. Irlicht, "Renewal-Process Approximation of a Stochastic Threshold Model for Electrical Neural Stimulation ," *J Comput Neurosci*, vol. 9, pp. 211-223, 2000.

[17] C. T. Kelley, *Iterative Methods for Optimization*, SIAM Frontiers in Applied Mathematics, 1999

[18] S. Kirkpatrick, C. D. Gelatt, M. P. Vecchi, "Optimization by Simulated Annealing," *Science*, Vol. 220, No. 4598, pp. 671-680, 1983.

[19] A. Corana, M. Marchesi, C. Martini, S. Ridella, "Minimizing Multimodal Functions of Continuous Variables with the "Simulated Annealing" Alogrithm," *ACM Transactions on Mathematical Software*, vol. 13, No. 3, pp. 262-280, 1987.

[20] M.L. Hines and N.T. Carnevale, "The NEURON simulation environment", *Neural Computat.* 9: 1179–1209, 1997.

[21] A. Destexhe, D. Contreras, M. Steriade, "Mechanisms Underlying the Synchronizing Action of Corticothalamic Feedback Through Inhibition of Thalamic Relay Cells," *J. Neurophysiol* vol. 79, pp. 999-1016, Feb. 1998.

[22] *http://senselab.med.yale.edu/senselab/modeldb/ShowModel.asp?model=3817*