


Fitting a recurrent dynamical neural network to neural spiking data: tackling the sigmoidal gain function issues

Reşat Özgür DORUK*

Department of Electrical and Electronics Engineering, Faculty of Engineering, Atılım University, Ankara, Turkey

Received: 03.08.2018

Accepted/Published Online: 04.12.2018

Final Version: 22.03.2019

Abstract: This is a continuation of a recent study (Doruk RO, Zhang K. Fitting of dynamic recurrent neural network models to sensory stimulus-response data. *J Biol Phys* 2018; 44: 449-469), where a continuous time dynamical recurrent neural network is fitted to neural spiking data. In this research, we address the issues arising from the inclusion of sigmoidal gain function parameters to the estimation algorithm. The neural spiking data will be obtained from the same model as that of Doruk and Zhang, but we propose a different model for identification. This will also be a continuous time recurrent neural network, but with generic sigmoidal gains. The simulation framework and estimation algorithms are kept similar to that of Doruk and Zhang so that we can have a solid base to compare the results. We evaluate the estimation performance in two different ways. First, we compare the firing rate responses of the original and the estimated model. We find that responses of both models to the same stimuli are similar. Secondly, we evaluate variations of the standard deviations of the estimates against a number of samples and stimulus parameters. They show a similar pattern to that of Doruk and Zhang. We thus conclude that our model serves as a reasonable alternative provided that firing rate is the response of interest (to any stimulus).

Key words: Neural spiking, maximum likelihood estimation, sigmoidal gain functions, parameter confounding

1. Introduction

1.1. General neuron modeling

Regardless of being mammalian or nonmammalian, a neuron is the fundamental component of the nervous system of a living being. Understanding the operation of these complicated structures is an integral part of theoretical and computational neuroscience research. Thus, neuron modeling has proven to be an indispensable tool in that research field.

Single compartmental models such as the classical Hodgkin–Huxley equation [2] describe the contribution of various ionic currents (sodium and potassium) to subthreshold behavior and to generation of spikes. Similar models such as [3–5] are all extensions of the Hodgkin–Huxley equation and incorporate the dynamics of calcium channels. It should be noted that these and also the original Hodgkin–Huxley equation are highly nonlinear. Thus, it is difficult to implement an efficient system identification procedure for them.

Several simplification attempts are seen in the literature. If emphasis is only given to dynamical behaviors such as phasic spiking, bursting, or spike frequency adaptation [6], then reduced order models such as Fitzhugh–Nagumo [7] or Hindmarsh–Rose [8] may be of concern. These models have fewer parameters to estimate.

*Correspondence: resat.doruk@atilim.edu.tr

When information coding and transmission are the primary research objectives, one can refer to models of moderate complexity. Linear-nonlinear cascades (or nonlinearity+linear filter combination) [9–11] and black-box models [12–15] fall under this category. The latter may reveal the statistical properties of a neural response to a given stimulus.

Stochasticity is crucial in neuron modeling [16]. Due to the random processes in ion channels, the reliability and precision of neural signaling are limited. Because of this, some statistical components should be integrated into the models. That will be a major factor in degradation of computational complexity and thus the deterministic component of the model should be chosen such that it has a good approximation capability.

In this respect, generic neural network models may be a reasonable choice. Static feedforward [17], dynamic feedforward [18], and recurrent neural networks [19] are well-known examples. They seem to have very few applications in computational neuroscience (see Section 1.2).

The recurrent neural networks have self-excitatory and self-inhibitory connections, which may be thought of as being equivalent to autapses (autaptic synapses). Neural behaviors due to autaptic connections attracted several researchers to the problem. One can refer to the references [20–22] as examples.

In addition, generic neural networks may have multilayered structures. With their universal approximation capability [23], those appear as a reasonable medium to model biological complexity [24, 25].

1.2. Neural spiking issue

The information transmitted through biological neural networks (especially the sensory subsystem) are hidden in the temporal locations of individual action potentials fired in a burst [26]. The phenomenon involving a series of action potentials (appearing in bursts) is called neural spiking. Due to the stochasticity of ion channels and synapses (Section 1.1), it is not a deterministic process [27].

First spike latency is a known phenomenon in this respect. One can refer to [28, 29] to investigate the effects of channel stochasticity on the first spike latency in Hodgkin–Huxley neurons.

Moreover, recent research suggests that the spiking activity of sensory neurons obeys an inhomogeneous Poisson process (IPP). Here, the event rate of an IPP is equivalent to the firing rate of the generated spikes [30].

Statistical identification techniques such as maximum likelihood (ML) [31] or maximum a posteriori (MAP) [32] estimation may be utilized to identify models from neural spiking data. DiMattina and Zhang [33, 34] applied these methods to train the parameters of a static feedforward neural network to describe the behavior of the auditory cortex in the brains of marmoset monkeys. In [1], one can see a theoretical study aiming at the estimation of the weights of an excitatory-inhibitory network from neural spiking data. One advantage of [1] over [33, 34] is that it takes the dynamical properties of a neuron into consideration. The models in those studies are rather computational and provide the firing rate of the excitatory neuron as output (which is vested in the spikes).

1.3. About this work

The purpose of this research is to address a major issue associated with the estimation of the parameters of sigmoidal gain functions in the recurrent neural network models of [1]. Those parameters are the maximum firing rate, threshold level, and slope. In [1], it is assumed that they are known. However in a realistic application,

these parameters are not known. In addition, a recent study [35] suggested that the algorithm presented in [1] is very inefficient if the sigmoidal gain function parameters are taken into consideration.

In this work, the problem is revisited. The neural spiking data are obtained from the simulation of the model in [1] with its nominal parameters. However, the sigmoidal gain functions of the original model in [1] are replaced by generic sigmoids with a predefined slope parameter. The same estimation procedures and stimulus (a truncated Fourier series) as that of [1] are applied to the problem.

As the data generator and the estimated models are different in form, we will not be able to talk about any true parameters. However, the accuracy of the estimation may be evaluated by comparing the firing rates obtained from the original data generator and from the estimated models. The firing rate outputs of the models against two different stimuli (square wave and Fourier series) will be compared.

The performances of estimation against the number of samples, the amplitude, and frequency and number of harmonic components of stimulus will also be investigated.

2. Materials and methods

2.1. Neural model in consideration

Like that of [1], the model considered in this research is also an excitatory-inhibitory recurrent neural network, as shown below:

$$\begin{aligned}\dot{x}_e &= -\beta_e [x_e + w_{ee}g(x_e) - w_{ei}g(x_i) + c_e u], \\ \dot{x}_i &= -\beta_i [x_i + w_{ie}g(x_e) - w_{ii}g(x_i) + c_i u],\end{aligned}\tag{1}$$

where x_e and x_i are dimensionless variables representing the activities of the excitatory ‘e’ and inhibitory ‘i’ units, respectively. The functions $g(x_e)$ and $g(x_i)$ are generic logistic sigmoid functions:

$$g(x) = \frac{1}{1 + e^{-\alpha x_k}},\tag{2}$$

where k is either e or i . The parameter α is a small scaling constant. It should be a small positive real number such as $\alpha = 0.001$. Eq. (2) is the major difference of our model from the corresponding one in [1]. In [1], sigmoids have three parameters (maximum firing rate, threshold, and slope) and they may have different parameters for excitatory and inhibitory units. In the original model, one thus has six parameters other than the network weights. These were assumed as known parameters in [1]. In this research, we will not have any tunable parameters other than the scaling factor α . It is not required to estimate this parameter. In fact, we preassign this factor to relax the estimation (to prevent unreasonably large values, etc.).

Since we are collecting neural spiking data, we will need to map the firing rate of the neurons in the network to its activities (i.e. to the variables x_e or x_i). In this work, we use the data from the excitatory neuron. Because of that, we will need to map the activity of the excitatory unit x_e in Eq. (1) to its firing rate r_e as shown below:

$$r_e = \frac{F_e}{1 + \exp(-\alpha x_e)}.\tag{3}$$

The above is also a logistic sigmoid function with additional parameter, which is the maximum firing rate of the excitatory neuron: F_e . In the above, we again have a small scaling factor similar to Eq. (2). It can also be chosen as $\alpha = 0.001$. There are a total of nine parameters to be estimated in the whole system defined by

Eqs. (1), (2), and (3). These are $\theta = [\beta_e, \beta_i, w_e, w_i, w_{ee}, w_{ei}, w_{ie}, w_{ii}, F_e]$. One can refer to Table 1 for their definitions and units. Similar to that of [1], the stimulus u will be modeled as a periodic signal such as a phased cosine Fourier series.

Table 1. Definitions and units of estimated parameters in Eqs. (1), (2), and (3). It should be noted that these are all positive.

Parameter	Definition	Unit
β_e	Excitatory unit time constant	s^{-1}
β_i	Inhibitory unit time constant	s^{-1}
w_{ee}	Self-excitation (autaptic) coefficient	None
w_{ii}	Self-inhibition (autaptic) coefficient	None
w_{ei}	Inhibitory to excitatory synaptic weight	None
w_{ie}	Excitatory to inhibitory synaptic weight	None
c_e	Excitatory presynaptic weight	None
c_i	Inhibitory presynaptic weight	None
F_e	Maximum firing rate of the excitatory unit	s^{-1}

The needed neural spiking data will be generated by a data generator model. As mentioned in Section 1.3, we will obtain these data from simulation of the original excitatory-inhibitory network in [1] with its nominal parameters. The details are discussed in Section 2.3.

2.2. Neural spiking and Poisson processes

2.2.1. Inhomogeneous Poisson processes

IPPs are stochastic point process models that are widely used in computational neuroscience. An IPP is related to the famous Poisson distribution [36] that defines the probability of detecting K occurrences in an interval of $[t, t + \Delta t)$ as:

$$\text{Prob}[N(t + \Delta t) - N(t) = k] = \frac{e^{-\lambda} \lambda^k}{k!}, \quad (4)$$

where the average number of occurrences (or the so-called event rate) λ is given by:

$$\lambda = \int_t^{t+\Delta t} r(\tau) d\tau. \quad (5)$$

In the above, $r(t)$ is the time-dependent event rate. In the context of neural spiking, it corresponds to the firing rate of the neuron of interest. In that case, K will be the number of expected spikes in the same time interval. IPPs may be compared to the local Bernoulli processes [37]. Let δt be a small time interval such that only one spike can fit (in one bin). If the instantaneous firing rate of the spiking neuron is $r(t)$, the probability that a spike may fire in $[t, t + \delta t)$ equals $r(t)\delta t$ and the probability that no spike will be seen in the same interval is $1 - r(t)\delta t$. Suppose that K spikes occur at $0 \leq t_1 \leq t_2 \leq \dots \leq t_K \leq T$. Assuming that these are independent, one can write the following:

$$p(t_1, t_2, \dots, t_K) = \exp\left(-\int_0^T r(t, u, \theta) dt\right) \prod_{k=1}^K r(t_k, u, \theta), \quad (6)$$

provided that $\delta t \rightarrow 0$. The above formulation describes how likely a particular spike train t_1, t_2, \dots, t_K is generated by a neural process with firing rate $r(t, u, \theta)$. It should be noted that firing rate $r(t, u, \theta)$ depends on the neuron parameters θ as well as the stimulus u .

2.2.2. Maximum likelihood methods and parameter estimation

Concerning the neural spiking processes modeled by IPPs, the likelihood as a function of neuron parameters θ and stimulus u is given by Eq. (6). However, this function involves only a single recording of spikes detected in $[0, T)$. We know from estimation theory that maximum likelihood estimation is asymptotically efficient, i.e. reaching the Cramér–Rao bound when the data sample size becomes larger. In order to improve the statistical content of the obtained data, one will need to collect multiple data elicited from multiple stimuli. Suppose that we have N_{it} independent stimuli that yielded N_{it} different spike trains and that the m th stimulus ($m = 1, \dots, N_{it}$) elicits a spike train with a total of K_m spikes in the time window $[0, T]$. Suppose also that the spike timings are given by $S_m = (t_1^{(m)}, t_2^{(m)}, \dots, t_{K_m}^{(m)})$. Using Eq. (6), one can evaluate the likelihood for the spike train S_m as:

$$p(S_m | \theta) = \exp\left(-\int_0^T r^{(m)}(t) dt\right) \prod_{k=1}^{K_m} r^{(m)}(t_k^{(m)}). \quad (7)$$

In the above, $r^{(m)}$ is the firing rate response to the m th stimulus. This function depends implicitly on the model parameters θ and on the stimulus. The left-hand side of Eq. (7) emphasizes the dependence on network parameter θ , which is convenient for parameter estimation. If the responses to each individual stimulus m are independent (this will be a reasonable assumption especially when the durations between consecutive stimuli are sufficiently large), the joint likelihood function for N_{it} spike trains can be written as shown below:

$$L(S_1, S_2, \dots, S_M | \theta) = \prod_{m=1}^M p(S_m | \theta). \quad (8)$$

To ease the computation, one is recommended to evaluate the natural logarithm of Eq. (8) and derive the log-likelihood function as:

$$l(S_1, S_2, \dots, S_M | \theta) = -\sum_{m=1}^M \int_0^T r^{(m)}(t) dt + \sum_{m=1}^M \sum_{k=1}^{K_m} \ln r^{(m)}(t_k^{(m)}). \quad (9)$$

The maximization of the above with respect to the parameter vector θ will yield the maximum likelihood estimate. That is:

$$\hat{\theta}_{ML} = \arg \max_{\theta} [l(S_1, S_2, \dots, S_M | \theta)]. \quad (10)$$

The above operation can be performed by MATLAB optimization routines such as `fmincon`. A zero lower bound should be assigned for each parameter so that positive values are obtained.

2.3. The data generator

Since the needed spiking data are obtained from simulations of the original model in [1], it will be convenient to present a summary of that in the proceeding sections.

2.3.1. Data generator model

The data generator is also a continuous time recurrent neural network model as shown below [1]:

$$\begin{aligned}\dot{v}_e &= -\beta_e [v_e + w_{ee}g_e(v_e) - w_{ei}g_i(v_i) + c_e u], \\ \dot{v}_i &= -\beta_i [v_i + w_{ie}g_e(v_e) - w_{ii}g_i(v_i) + c_i u],\end{aligned}\tag{11}$$

where v_e and v_i represent the activities of the excitatory and inhibitory units, respectively. They can be thought of as representing the membrane potential. The critical difference from Eq. (1) is that it has different sigmoidal gain functions ($g_e(v_e)$ and $g_i(v_i)$). These are shown below:

$$g_j(v_j) = \frac{\Gamma_j}{1 + \exp(-a_j(v_j - h_j))},\tag{12}$$

with j being either ‘ e ’ for excitatory units or ‘ i ’ for inhibitory units. Since we collect the spiking data from the excitatory unit, it will be convenient to rewrite its firing rate r_e as:

$$r_e = g_e(v_e) = \frac{\Gamma_e}{1 + \exp(-a_e(v_e - h_e))}.\tag{13}$$

The true values of parameters in Eqs. (11) and (12) are presented in Tables 2 and 3.

Table 2. The true values of the parameters of the network model in Eq. (11). These will be used in the simulations to generate neural spiking data.

Parameter	β_e	β_i	w_e	w_i	w_{ee}	w_{ei}	w_{ie}	w_{ii}
Value	50	25	1.0	0.7	1.2	2.0	0.7	0.4

Table 3. The parameters of the sigmoidal gain functions $g_j(V)$ in Eq. (12) for the excitatory (e) and inhibitory (i) units of the data generator model of Eq. (11). These will be used in the simulations to generate neural spiking data.

Parameter	Γ_e	a_e	h_e	Γ_i	a_i	h_i
Value	100	0.04	70	50	0.04	35

2.3.2. Spike generation for data collection

The data will be representative of in vivo neural spiking data. Because of that, they should be generated by a reliable algorithm.

Since neuronal spiking activity often has Poisson-like noise (see Section 2.2), for a better comparison with extracellular neurophysiological recording data, we consider spike trains generated by an IPP. In other words, we will simulate an IPP with r_e as its event rate.

There are several algorithms to simulate an IPP. The local Bernoulli approximation [37] (also discussed in Section 2.2), thinning [36], and time scale transformation [38] can be shown as examples.

The approach based on the local Bernoulli approximation is a reasonable choice when the neuron models are integrated by discrete solvers such as Euler or Runge–Kutta methods. However, one should keep the time bins very narrow (such as $\delta t = 1$ ms) so that the statistics of the spikes are closer to those of an IPP.

In this work, we also implement local Bernoulli approximation. One can see a summary of the related algorithm below [1]:

1. Given the firing rate of a neuron as $r(t)$
2. Find the probability of firing at time t_i by evaluating $p_i = r(t_i)\delta t$ where δt is the integration interval. It should be a small real number such as 1 ms.
3. Draw a random number $x_{rand} = U[0, 1]$, which is uniformly distributed in the interval $[0, 1]$. Here, U stands for a uniform distribution.
4. If $p_i > x_{rand}$ fire a spike at $t = t_i$, else do nothing.
5. Collect spikes as $S = [t_1, \dots, t_{N_s}]$, where N_s will be the total number of spikes collected from one simulation.

2.4. Stimulus model

Like that of [1], we will have a periodic phased cosine Fourier series stimulus, which is defined as:

$$u = \sum_{n=1}^{N_U} A_n \cos(\omega_n t + \phi_n), \quad (14)$$

where A_n is the amplitude, $\omega_n = 2\pi f_0 n$ is the frequency (in rad/σ), and ϕ_n is the phase (in radians) of the n th Fourier component in the above equation. Here the amplitude A_n and the base frequency f_0 (in Hz) are fixed but the phase ϕ_n will be randomly drawn from $U[-\pi, \pi]$ (i.e. uniformly distributed between $-\pi$ and π). The amplitude parameter A_n will also be fixed as $A_n = A_{\max}$ (for all mode n).

One can see the variations of Eq. (14) for different configurations in Figure A (see the explanation of the figure for details). Similarly, one can see the typical firing rate response patterns of the data generator model in Eqs. (11), (12), and (13) in Figure B. Here, the model is simulated with the parameters in Tables 2 and 3 against the stimuli in Figure A. Finally, in Figure C one can see the neural spiking responses of the excitatory unit against the stimuli in Figure A.

3. Application

3.1. Description of the example problem

We can summarize the example problem as shown in the following:

1. A single run of simulation will last for $T_f = 3$ s.
2. The data generator neuron model in Eqs. (11), (12), and (13) will be simulated with the parametric values given in Table 2 and Table 3. The firing rate of the excitatory unit is stored as $r_e^m(t)$, where m is an index for the current trial of simulation.
3. Using the approach presented in Section 2.3.2, the spike train S_m of the m th trial is generated from the firing rate $r_e^m(t)$. The number of spikes will be K_m at the m th trial.
4. Repeat the simulation N_{it} times to collect several statistically independent spike trains.

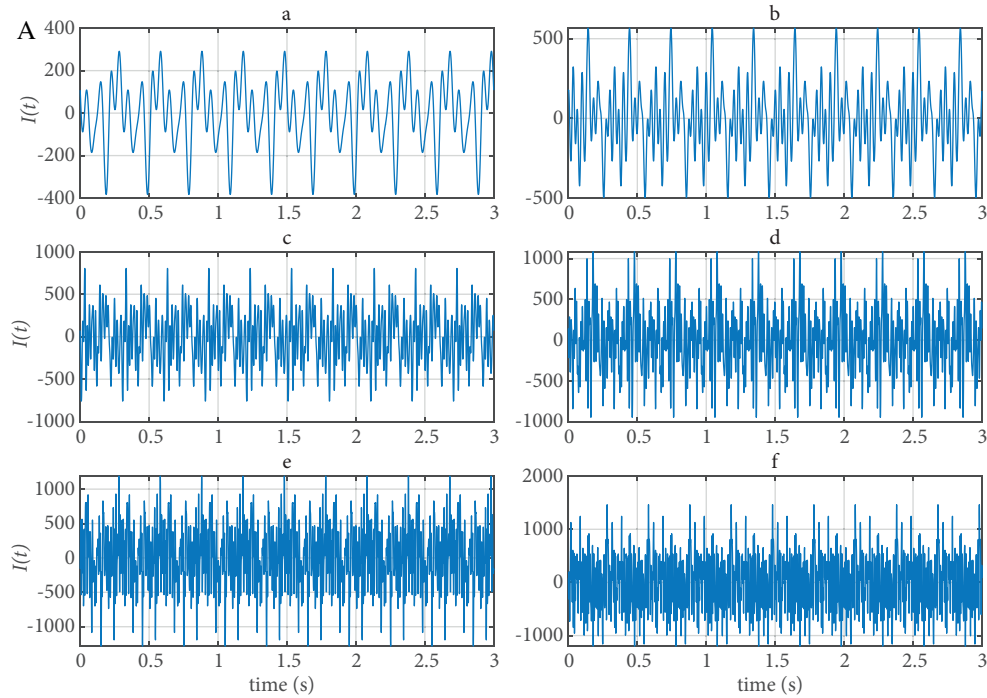


Figure. A The variation of cosine stimulus for different number of components N_U . The sizes are a) $N_U = 5$, b) $N_U = 10$, c) $N_U = 20$, d) $N_U = 30$, e) $N_U = 40$, and f) $N_U = 50$. The amplitude parameters are $A_n = 100$ and $f_0 = 3.333$ Hz, and ϕ_n are randomly assigned from a set uniformly distributed between $[-\pi, \pi]$. As the number of components increases, the magnitude of the stimulus also increases.

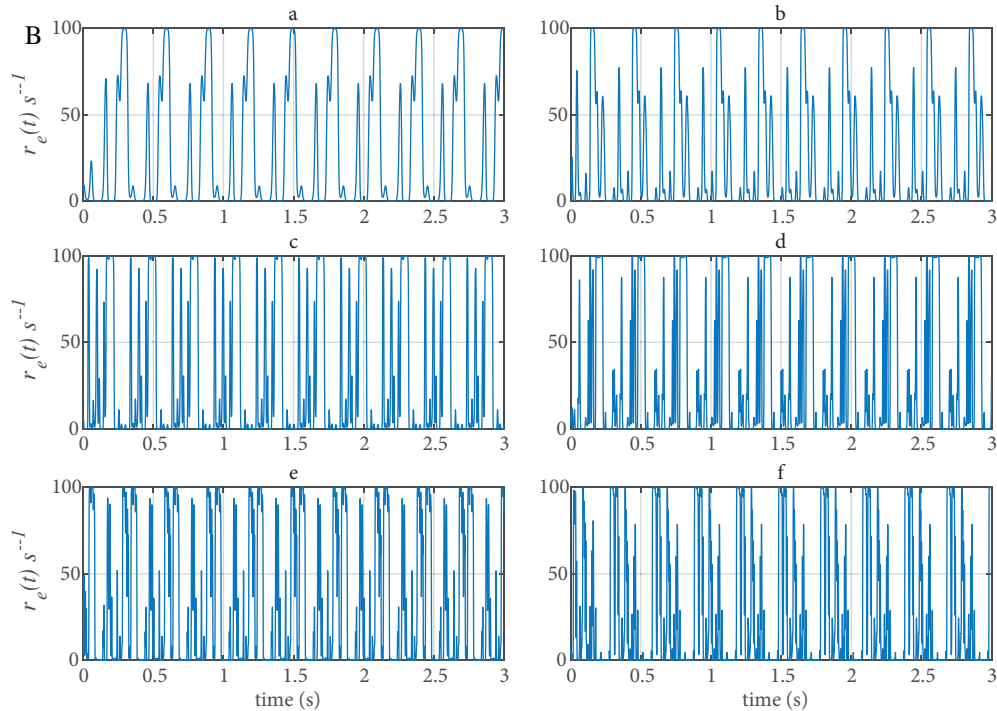


Figure. B The variation of the firing rate of the excitatory unit against the stimuli shown in Figure A. Note that the firing rates cannot exceed the upper limit imposed by the maximum firing rate parameter F_e , which equals 100 here. However, since the stimulus amplitude stays at larger values when one has a larger N_U , corresponding firing rates have more jumps to regions near the maximum limit set by F_e .

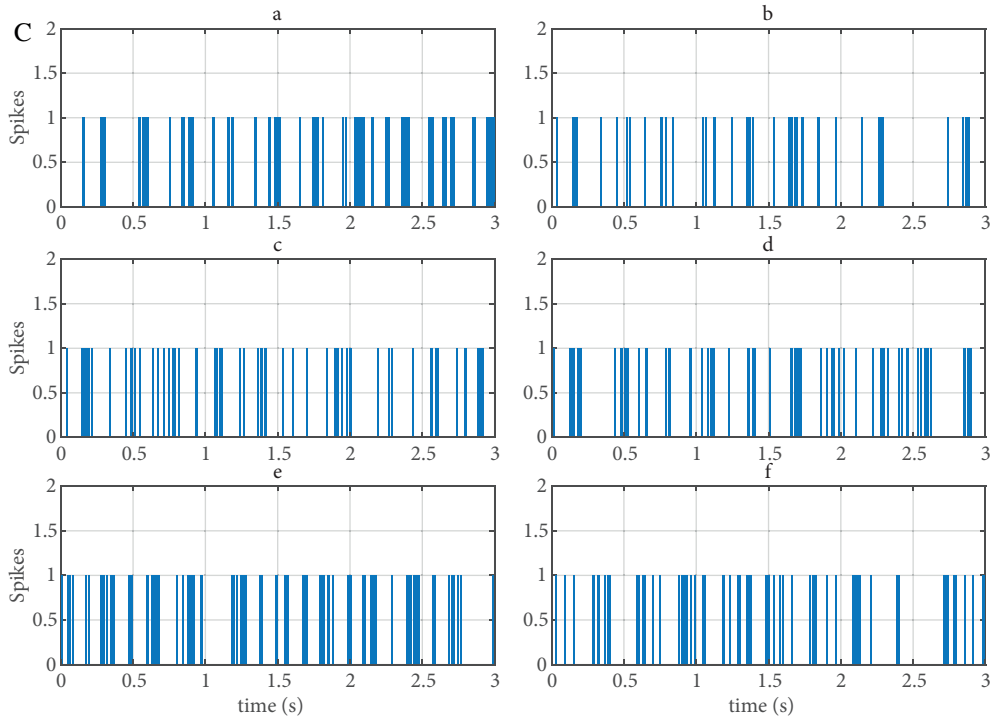


Figure C. Excitatory neural spiking against the stimuli shown in Figure A. Here the spikes are treated in a binary fashion, logged as a ‘1’ when a spike exists and ‘0’ when there is no spike. As expected, larger numbers of spikes will be seen in the regions where the firing rate stays at larger values (or with more jumps to high rate regions).

5. The neural spiking data needed by Eq. (9) will be obtained in the 4th step. However, the firing rate $r_e^m(t)$ in Eq. (9) should be computed at the current iteration of the optimization.
6. Run an optimization algorithm (such as `fmincon`) on the joint likelihood function of Eq. (9) to obtain the maximum likelihood estimates of the parameters (θ_{ML} in Eq. (10)).

3.2. Optimization algorithm

To perform a maximum likelihood estimation (i.e. the problem defined in Eq. (10)), one needs to use an optimizer. In this work, we will utilize MATLAB’s `fmincon` routine. This is a gradient-based local optimizer that allows the user to set a bound on the solution. A zero bound can ensure positively valued estimates. In addition, `fmincon` can evaluate the gradient of the objective function using finite difference approximations.

Of course, `fmincon` is not the only optimization routine available from MATLAB. There are other alternatives such as `fminunc`. That is similar to `fmincon` but it does not allow any bounds to be set on the solution. Thus, it is not applicable to our problem, although it is faster than `fmincon` (no lower bounds can be set).

In addition to those, there are gradient-free techniques such as pattern search, simulated annealing, and genetic algorithms. Although elimination of gradients may be considered as an advantage, these routines appear to have slower convergence in multivariable problems.

The computational complexity of the problem is considerably high. Thus, additional improvement of the optimization performance is necessary. That is obtained by using high-performance computing (HPC) systems.

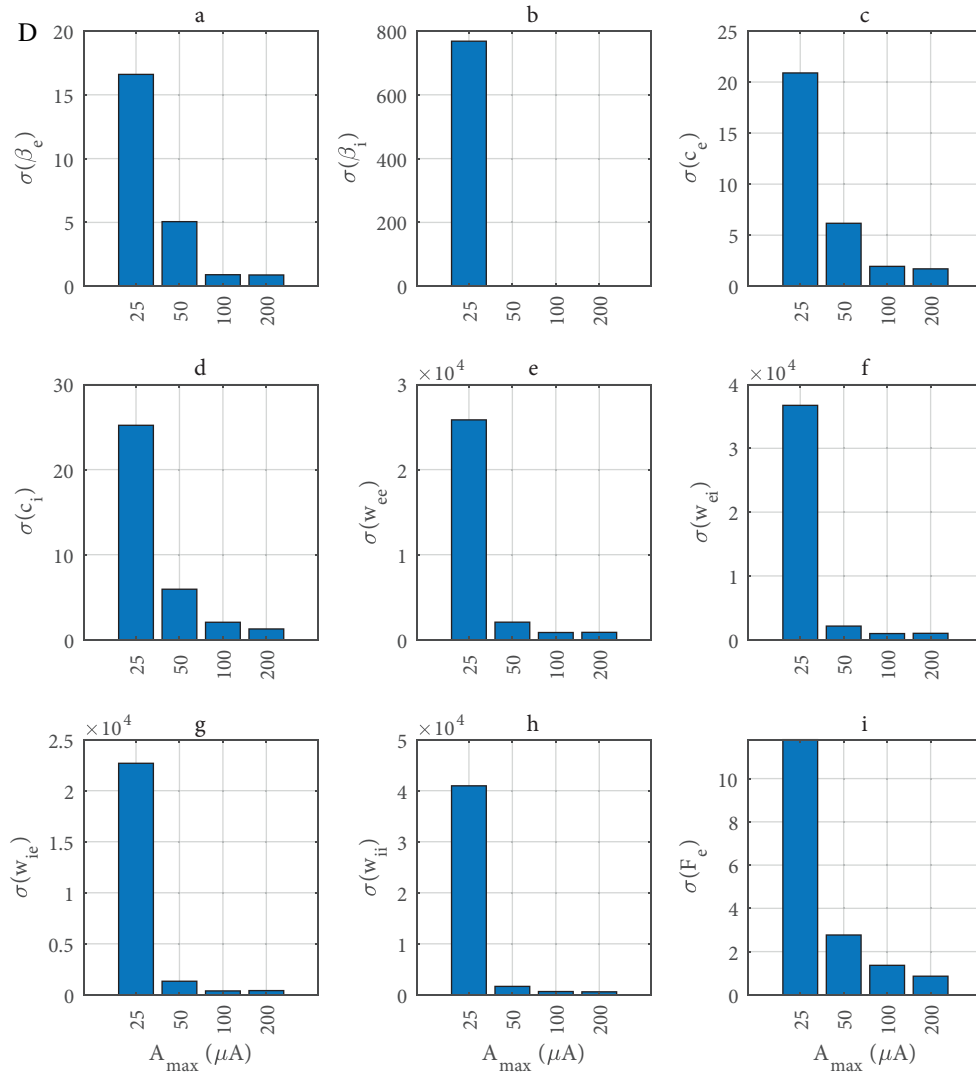


Figure D The variations of the standard deviations of the estimated parameters against increasing sample size N_{it} . The settings of the other simulation parameters are as follows: $N_U = 5$, $A_{\max} = 100$, and $f_0 = 3.3333$ Hz. As expected, the standard deviations decrease with increasing number of samples. This is a fundamental property of maximum likelihood estimation.

3.3. Simulation scenarios

The information related to simulation scenarios is presented in Table 4.

Table 4. Typical data related to a nominal simulation scenario.

Parameter	Symbol	Value
Simulation time	T_f	3 s
Number of trials	N_{it}	25, 50, 100, 200, 400
# of components in stimulus	N_U	5, 10, 20
Method of optimization	N/A	Interior-point gradient descent (<code>fmincon</code>)
# of parameters estimate	$\text{Size}(\theta)$	9
Stimulus amplitude (μA)	A_{\max}	25, 50, 100, 200
Base frequency	f_0	$1/3$, 1, $7/3$, $10/3$, 5 Hz

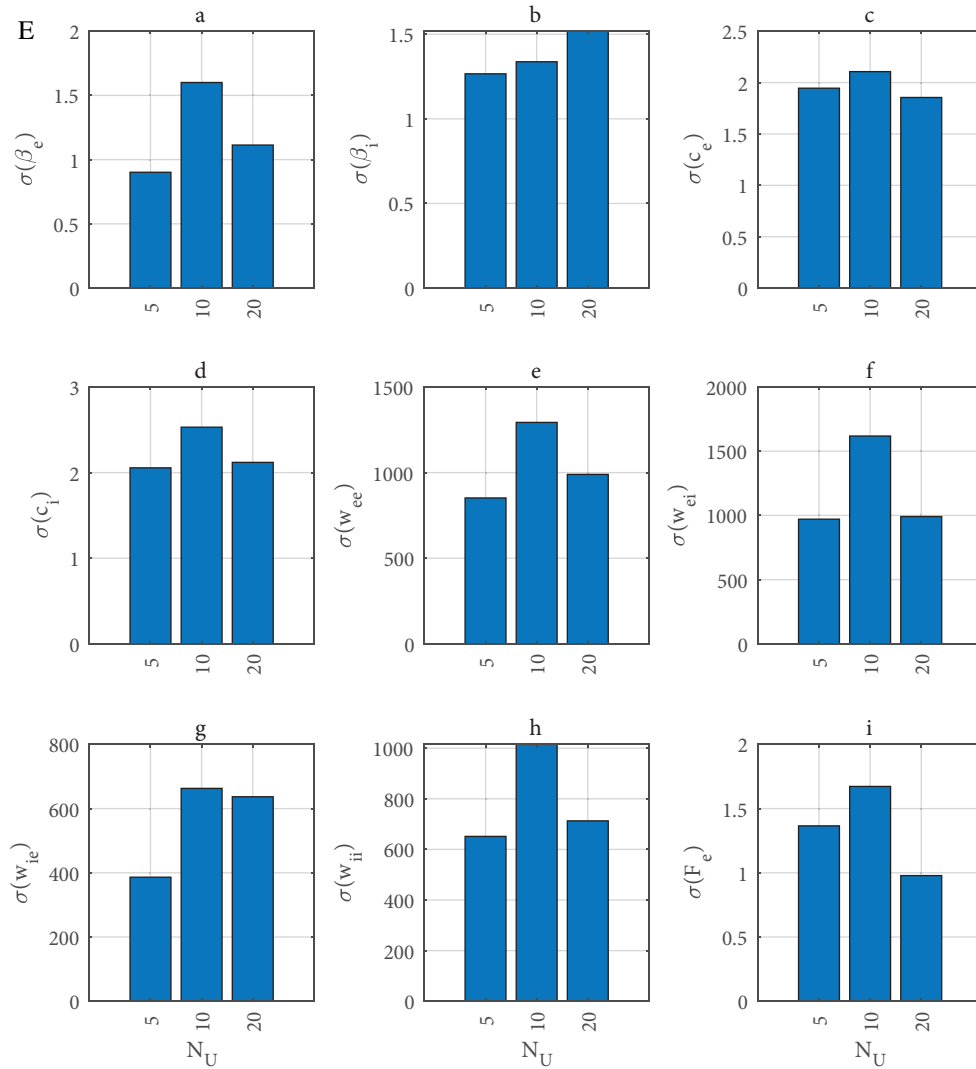


Figure. E The variations of the standard deviations of the estimated parameters against stimulus component size N_U . The settings of the other simulation parameters are as follows: $N_{it} = 100$, $A_{\max} = 100$, and $f_0 = 3.3333$ Hz. We are unable to see a definite pattern associated with the changing stimulus component count, but one can say that $N_U = 5$ is a reasonable choice.

In order to evaluate the estimation performance under different conditions, it will be convenient to analyze the effects of changes in the number of samples N_{it} , stimulus component size N_U , amplitude level A_{\max} , and base frequency f_0 . That is why we have more than one entry for those simulation parameters in Table 4.

The initial conditions of the excitatory and inhibitory activity variables in Eq. (1) will be assumed as $x_e(0) = 0$ and $x_i(0) = 0$. This is a reasonable choice as we do not have any information about them.

4. Results

In this section, one will be able to examine the results of our example problem. The maximum likelihood estimates (θ_{ML}) of the parameters (θ) of Eqs. (1), (2), and (3) are obtained by maximizing Eq. (8) using MATLAB's `fmincon` routine. The relevant results fall into three categories, which are shown below:

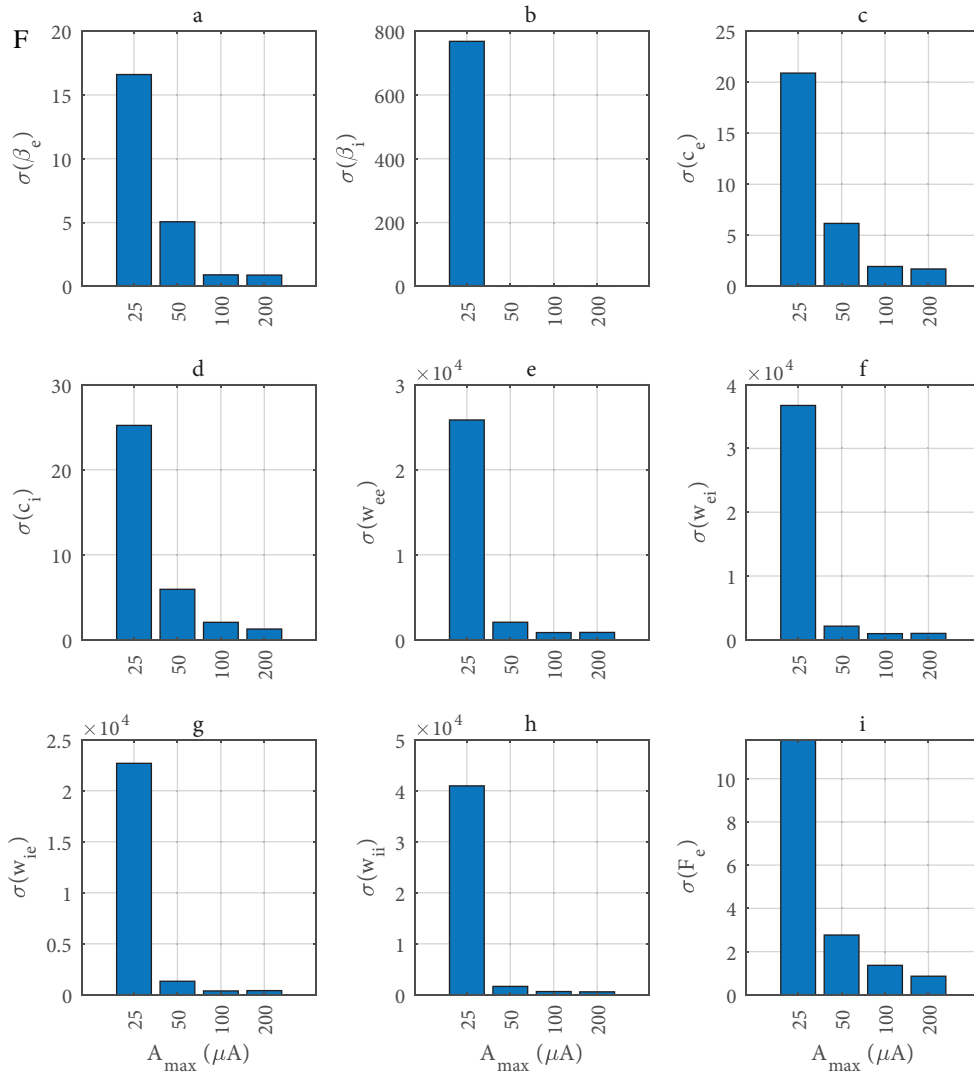


Figure. F The variations of the standard deviations of the estimated parameters against maximum amplitude parameter A_{\max} . The settings of the other simulation parameters are as follows: $N_{it} = 100$, $N_U = 5$, and $f_0 = 3.3333$ Hz. In general, A_{\max} seems to have a similar effect as N_{it} . However, for some parameters such as w_{ee} large values like $A_{\max} = 200$ show no improvement, so $A_{\max} = 100$ should be a good compromise.

1. The variations of mean estimated values of θ (θ_{ML}) against varying sample size N_{it} , amplitude level A_{\max} , stimulus component size N_U , and base frequency f_0 (see Section 4.1).
2. The variations of standard deviations of the estimated parameters against varying sample size N_{it} , amplitude level A_{\max} , stimulus component size N_U , and base frequency f_0 (see Section 4.2).
3. The variations of the mean square error between excitatory firing rates (r_e) of the data generator (Eqs. (11) and (12)) and the estimated (Eqs. (1), (2), and (3)) models. This is required to validate the estimation's accuracy as we cannot know the true values of the parameters of Eqs. (1), (2), and (3) (see Section 4.3).

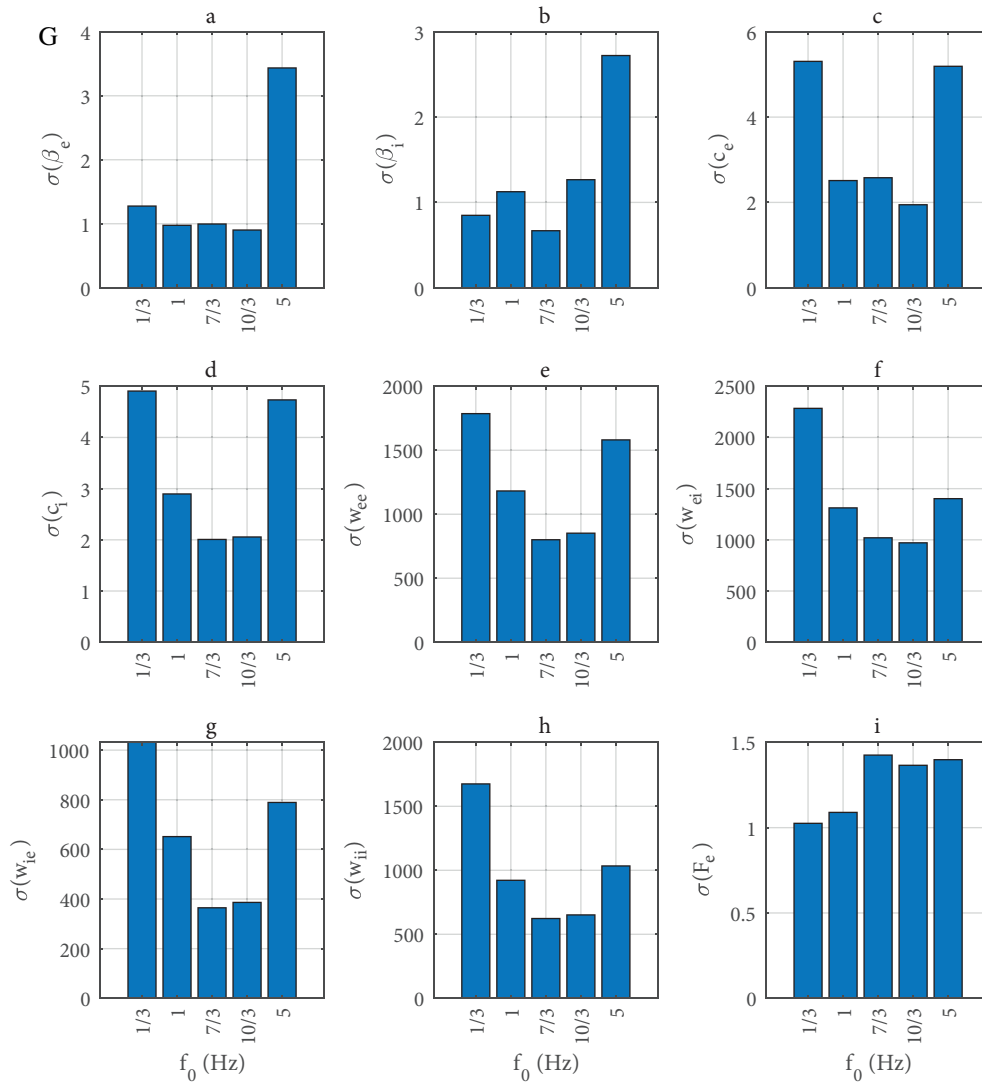


Figure. G The variations of the standard deviations of the estimated parameters against base frequency f_0 . The settings of the other simulation parameters are as follows: $N_{it} = 100$, $N_U = 5$, and $A_{max} = 5$. It appears that dependency on base frequency does not show a definite pattern. However, the deviations are smaller in the midrange. As a result, $f_0 = 3.3333$ Hz is a good choice of base frequency.

4.1. Mean values obtained from maximum likelihood

One can see the variations of the maximum likelihood estimates of the parameters (θ_{ML}) against sample size parameter N_{it} in Table 5.

4.2. Change in the standard deviation of the estimates

One can see the variations of the standard deviations of the estimates (θ_{ML}) against the sample size N_{it} , stimulus component size N_U , maximum amplitude parameter A_{max} , and stimulus base frequency f_0 in Figures D, E, F and G respectively.

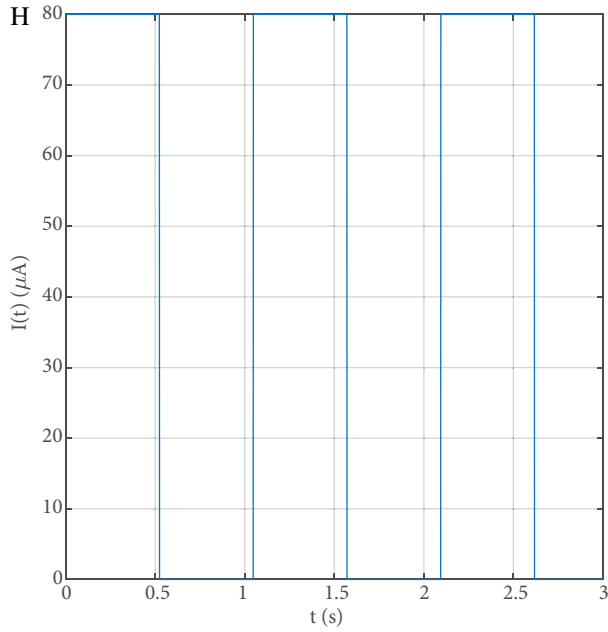


Figure. H Square wave test stimulus. This is a discontinuous test signal.

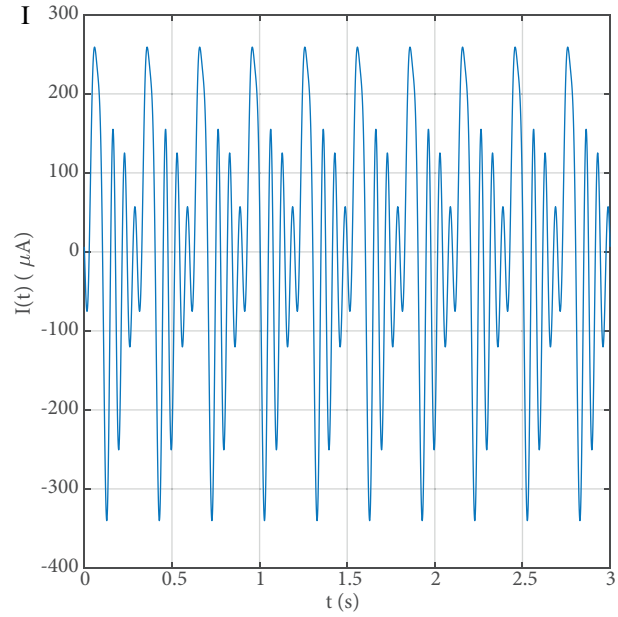


Figure. I Fourier series test stimulus. This is a smooth test signal.

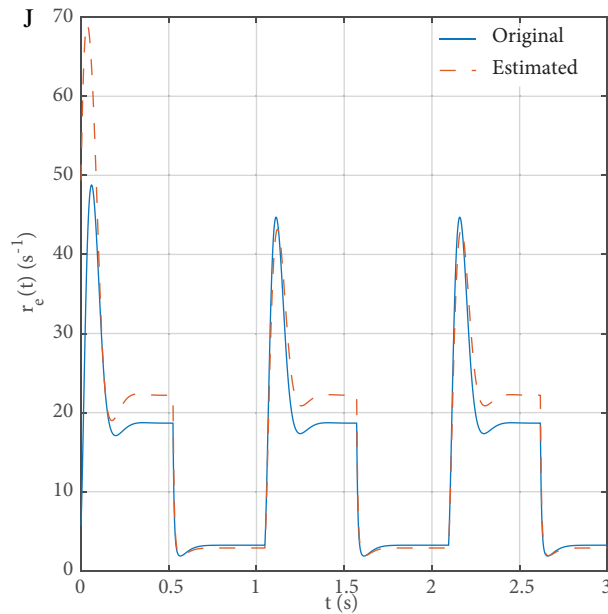


Figure. J Response of Eq. (1) to square wave stimulus in Figure. The model parameters are taken from the last row of Table 5. The responses show a small deviation at the discontinuities of the stimulus. However, the responses are synchronized most of the time.

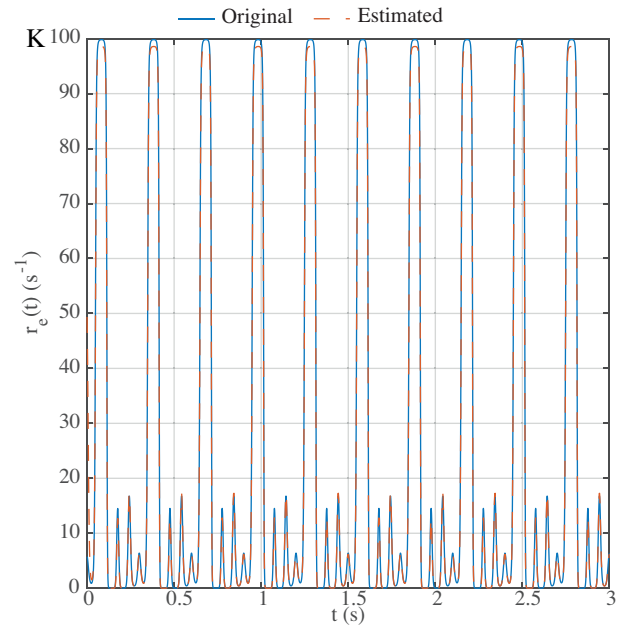


Figure. K Response of Eq. (1) to square wave stimulus in Figure. The model parameters are taken from the last row of Table 5. Almost the same response is received from the data generator and estimated model.

Table 5. The variations of the maximum likelihood estimates (θ_{ML}) against sample size N_{it} are presented. The settings of other simulation parameters are as follows: $N_U = 5$, $A_{max} = 100$, and $f_0 = 3.3333$ Hz.

N_{it}	β_e	β_i	w_e	w_i	w_{ee}	w_{ei}	w_{ie}	w_{ii}	F_e
25	38.26	25.48	53.98	20.48	7725.53	13883.68	2796.59	4171.99	98.58
50	36.23	25.94	55.60	20.64	7640.84	13923.25	2637.48	4005.38	98.87
100	36.28	26.04	55.97	20.63	7427.70	13666.09	2533.39	3844.03	98.76
200	36.17	26.42	55.83	20.97	7539.17	13669.48	2555.44	3859.42	98.47
400	36.23	26.42	55.68	20.77	7252.09	13256.03	2428.22	3615.60	98.72

Table 6. The variations of the maximum likelihood estimates (θ_{ML}) against stimulus component size N_U are presented. The settings of other simulation parameters are as follows: $N_{it} = 100$, $A_{max} = 100$, and $f_0 = 3.3333$ Hz.

N_U	β_e	β_i	w_e	w_i	w_{ee}	w_{ei}	w_{ie}	w_{ii}	F_e
5	36.28	26.04	55.97	20.63	7427.70	13666.09	2533.39	3844.03	98.76
10	37.64	26.61	53.21	18.88	6858.12	12768.65	2254.27	3339.32	99.39
20	37.81	26.76	53.29	19.56	7131.97	13046.52	2087.44	3300.90	99.01

Table 7. The variations of the maximum likelihood estimates (θ_{ML}) against maximum amplitude A_{max} are presented. The settings of other simulation parameters are as follows: $N_{it} = 100$, $N_U = 5$, and $f_0 = 3.3333$ Hz.

A_{max}	β_e	β_i	w_e	w_i	w_{ee}	w_{ei}	w_{ie}	w_{ii}	F_e
25	53.21	515.45	54.44	27.31	20896.08	38946.13	18011.48	33635.76	51.18
50	35.00	22.94	58.42	25.15	9653.59	18991.63	4407.25	7983.17	95.89
100	36.28	26.04	55.97	20.63	7427.70	13666.09	2533.39	3844.03	98.76
200	40.55	28.73	49.75	14.67	6537.35	11855.27	1528.09	2215.13	99.15

Table 8. The variations of the maximum likelihood estimates (θ_{ML}) against base frequency f_0 are presented. The settings of other simulation parameters are as follows: $N_{it} = 100$, $N_U = 5$, and $A_{max} = 100$ Hz.

f_0	β_e	β_i	w_e	w_i	w_{ee}	w_{ei}	w_{ie}	w_{ii}	F_e
1/3	30.84	26.69	69.63	31.65	8864.84	15372.76	3447.75	5210.14	98.58
1	34.69	27.65	59.15	23.10	7217.08	12867.85	2378.26	3385.87	98.97
7/3	36.48	26.71	56.04	19.90	6310.13	11976.28	1901.61	2735.95	98.46
10/3	36.28	26.04	55.97	20.63	7427.70	13666.09	2533.39	3844.03	98.76
5	38.36	23.57	52.57	18.71	7088.79	14886.53	2462.94	4750.91	99.14

4.3. Comparing the outputs of original and estimated models

In order to validate the success of our work, we will need to compare the firing rate outputs of the data generator (Eqs. (11), (12), and (13)) and estimated models (Eqs. (1), (2), and (3)).

In Figure H and Figure I, one can see the test stimuli that are used to examine the firing rate outputs of the two models. The former is a square wave stimulus, whereas the latter is a truncated Fourier series stimulus that has smoother edges.

The responses are compared by plotting the responses of both models to the same stimulus in the same

figure. The response of the estimated model is evaluated for the parameters obtained in the last row of Table 5. The response of the data generator is evaluated for its true parameters to the same stimulus. One can see the related results in Figure J and K, respectively.

5. Discussion

5.1. Summary

In this research, we presented a theoretical study of model fitting to noisy stimulus/response data obtained from a generic sensory neural network. In general, the purpose is similar to that of [1]. The main differences were:

1. We desire a more generic model that has a smaller number of parameters. It will model the relationship between stimulus and response of the neuron (i.e. the firing rate of the excitatory unit).
2. In [35], it was found that inclusion of sigmoidal gain function parameters in the estimation problem leads to a very inefficient estimator. For example, no definite patterns of variance are seen.

Thus, we desired a model that does not have any unknown parameters in its sigmoidal gain functions. That will decrease the level of parameter confounding and thus we expect a more efficient estimation. The neural spiking data are collected from the results of simulations of the neural network model presented in [1]. Since the data generator and estimated models are different, the validity of the approach is evaluated by comparing their firing rate responses. In addition, the estimation performance is evaluated by comparing the standard deviations of the estimates.

5.2. Discussion of the results

In Tables 5–8, one can see the variations of maximum likelihood estimates of the parameters of Eqs. (1), (2), and (3) against varying sample size N_{it} , component count N_U , maximum amplitude A_{\max} , and base frequency f_0 of the stimulus. The values seem to remain in a narrow range. One caveat appears when $A_{\max} = 25$. The values of parameters w_{ei} and w_{ie} appear different from the ones obtained when the stimulus amplitude is larger than $A_{\max} = 25$. This result verifies that the nominal selection of $A_{\max} = 100$, $N_U = 5$, and $f_0 = 10/3$ is a good compromise.

The results associated with the variations of standard deviations against the simulation parameters are available in Figures D, E, F and G. No unexpected situation exists in these results. The standard deviations of the estimates decrease with increasing sample size N_{it} . This occurrence is similar to that of [1].

There is no definite pattern associated with changing stimulus component size N_U . Thus, there is no need to choose large N_U values that bring unnecessary computational complexity.

Concerning the stimulus amplitude A_{\max} , one can say that lower amplitudes should be avoided. Figure F suggests that choosing a value like $A_{\max} = 25$ may adversely affect the estimation performance. The same graphical results also suggest that a selection of $100 \leq A_{\max} \leq 200$ should be a good compromise.

The changes in base frequency may also degrade the estimation performance. Figure G suggests that midrange frequencies ($7/3 \leq f_0 \leq 10/3$ Hz) should be preferred.

Concerning the comparison of firing rate responses, we can make the following comments:

1. One can say that under a smooth stimulus like a Fourier series, the data generator and estimated models generate almost the same firing rate response (Figure K).

2. When a discontinuous stimulus like in Figure H is applied, the responses elicited by both models have a small deviation at the discontinuities in Figure K. However, the responses are mostly close to each other.

Based on the above findings one can state that the estimation algorithm completed its duty quite successfully. With the completion of this work, we can show that one is able to develop a neural network model from noisy stimulus/response data without any factors that degrade the efficiency of estimation (like the parametrized sigmoidal gain functions).

Acknowledgment

The computational facilities needed by this research were provided by the TR-Grid/TRUBA framework (www.truba.gov.tr), operated by the National Academic Center of Computing (ULAKBİM) of the Scientific and Technological Research Council of Turkey (TÜBİTAK).

References

- [1] Doruk RO, Zhang K. Fitting of dynamic recurrent neural network models to sensory stimulus-response data. *J Biol Phys* 2018; 44: 449-469.
- [2] Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J Physiol-London* 1952; 117: 500. <https://doi.org/10.1113>.
- [3] Morris C, Lecar H. Voltage oscillations in the barnacle giant muscle fiber. *Biophys J* 1981; 35: 193-213.
- [4] Ermentrout B, Pascal M, Gutkin B. The effects of spike frequency adaptation and negative feedback on the synchronization of neural oscillators. *Neural Comput* 2001; 13: 1285-1310.
- [5] Booth V, Rinzel J, Kiehn O. Compartmental model of vertebrate motoneurons for Ca²⁺-dependent spiking and plateau potentials under pharmacological treatment. *J Neurophysiol* 1997; 78: 3371-3385.
- [6] Koch C. *Biophysics of Computation: Information Processing in Single Neurons*. Oxford, UK: Oxford University Press, 2004.
- [7] FitzHugh R. Impulses and physiological states in theoretical models of nerve membrane. *Biophys J* 1961; 1: 445-466.
- [8] Hindmarsh JL, Rose R. A model of neuronal bursting using three coupled first order differential equations. *Proc R Soc Lond B* 1984; 221: 87-102.
- [9] Rust NC, Schwartz O, Movshon JA, Simoncelli EP. Spatiotemporal elements of macaque v1 receptive fields. *Neuron* 2005; 46: 945-956.
- [10] Hosoya T, Baccus SA, Meister M. Dynamic predictive coding by the retina. *Nature* 2005; 436: 71. <https://doi.org/10.1016>
- [11] Mante V, Frazor RA, Bonin V, Geisler WS, Carandini M. Independence of luminance and contrast in natural scenes and in the early visual system. *Nat Neurosci* 2005; 8: 1690. <https://doi.org/10.1038/nn1556>.
- [12] Borst A, Theunissen FE. Information theory and neural coding. *Nat Neurosci* 1999; 2: 947. <https://doi.org/10.1038/14731>.
- [13] Barlow H. Possible principles underlying the transformation of sensory messages. In: Rosenblith W, editor. *Sensory Communication*. Cambridge, MA, USA: MIT Press, 1959, pp. 217-234.
- [14] Fairhall AL, Lewen GD, Bialek W, van Steveninck RRR. Efficiency and ambiguity in an adaptive neural code. *Nature* 2001; 412: 787. <https://doi.org/10.1038/35090500>.
- [15] Hassenstein B, Reichardt W. Systemtheoretische analyse der zeit-, reihenfolgen-und vorzeichenauswertung bei der bewegungsperzeption des rüsselkäfers chlorophanus. *Z Naturforsch B* 1956; 11: 513-524 (in German).

- [16] Schneidman E, Freedman B, Segev I. Ion channel stochasticity may be critical in determining the reliability and precision of spike timing. *Neural Comput* 1998; 10: 1679-1703.
- [17] Haykin S. *Neural Networks and Learning Machines*. 3rd ed. Upper Saddle River, NJ, USA: Pearson, 2009.
- [18] Arulampalam G, Bouzerdoum A. A generalized feedforward neural network architecture for classification and regression. *Neural Networks* 2003; 16: 561-568.
- [19] Funahashi KI, Nakamura Y. Approximation of dynamical systems by continuous time recurrent neural networks. *Neural Networks* 1993; 6: 801-806.
- [20] Guo DQ, Chen MM, Perc M, Wu SD, Xia C, Zhang YS, Xu P, Xia Y, Yao DZ. Firing regulation of fast-spiking interneurons by autaptic inhibition. *Europhys Lett* 2016; 2016: 114. <https://dx.doi.org/10.1209/0295-5075/114/30001>.
- [21] Guo DQ, Wu SD, Chen MM, Perc M, Zhang YS, Ma JL, Cui Y, Xu P, Xia Y, Yao DZ. Regulation of irregular neuronal firing by autaptic transmission. *Sci Rep* 2016; 6: 26096. <https://dx.doi.org/10.1038/srep26096>.
- [22] Yilmaz E, Ozer M, Baysal V, Perc M. Autapse-induced multiple coherence resonance in single neurons and neuronal networks. *Sci Rep* 2016; 6: 30914. <https://dx.doi.org/10.1038/srep30914>.
- [23] Schäfer AM, Zimmermann HG. Recurrent neural networks are universal approximators. In: *International Conference on Artificial Neural Networks*. pp. 632-640.
- [24] Gosak M, Markovic R, Dolensek J, Rupnik MS, Marhl M, Stozer A, Perc M. Network science of biological systems at different scales: a review. *Phys Life Rev* 2018; 24: 118-135. .
- [25] Jalili M, Perc M. Information cascades in complex networks. *J Complex Networks* 2017; 5: 665-693. .
- [26] Rieke F, Warland D, Bialek W. Coding efficiency and information rates in sensory neurons. *Europhys Lett* 1993; 22: 151. <https://doi.org/10.1209/0295-5075/22/2/013>.
- [27] Herz AV, Gollisch T, Machens CK, Jaeger D. Modeling single-neuron dynamics and computations: a balance of detail and abstraction. *Science* 2006; 314: 80-85.
- [28] Ozer M, Uzuntarla M, Perc M, Graham LJ. Spike latency and jitter of neuronal membrane patches with stochastic Hodgkin-Huxley channels. *J Theor Biol* 2009; 261: 83-92. <https://dx.doi.org/10.1016/j.jtbi.2009.07.006>.
- [29] Uzun R, Ozer M, Perc M. Can scale-freeness offset delayed signal detection in neuronal networks? *Europhys Lett* 2014; 2014: 105. <https://dx.doi.org/10.1209/0295-5075/105/60002>.
- [30] Shadlen MN, Newsome WT. Noise, neural codes and cortical organization. *Curr Opin Neurol* 1994; 4: 569-579.
- [31] Myung IJ. Tutorial on maximum likelihood estimation. *J Math Psychol* 2003; 47: 90-100.
- [32] Hsiao T. Identification of time-varying autoregressive systems using maximum a posteriori estimation. *IEEE T Signal Process* 2008; 56: 3497-3509.
- [33] DiMattina C, Zhang K. Active data collection for efficient estimation and comparison of nonlinear neural models. *Neural Comput* 2011; 23: 2242-2288.
- [34] DiMattina C, Zhang K. Adaptive stimulus optimization for sensory systems neuroscience. *Front Neural Circuit* 2013; 7: 101. <https://doi.org/10.3389/fncir.2013.00101>.
- [35] Doruk O, Zhang K. An attempt to fit all parameters of a dynamical recurrent neural network from sensory neural spiking data. *PeerJ Preprints* 2018; 6: e27015v1. <https://dx.doi.org/10.7287/peerj.preprints.27015v1>.
- [36] Lewis PA, Shedler GS. Simulation of nonhomogeneous Poisson processes by thinning. *Nav Res Logist Q* 1979; 26: 403-413.
- [37] Eden UT. Point process models for neural spike trains. *Neural Signal Processing: Quantitative Analysis of Neural Activity* 2008; 2008: 45-51.
- [38] Klein RW, Roberts SD. A time-varying Poisson arrival process generator. *Simulation* 1984; 43: 193-195.