COMMUN. MATH. SCI.
Vol. 17, No. 5, pp. 1313–1331

FAST ALGORITHMS FOR SIMULATION OF NEURONAL DYNAMICS BASED ON THE BILINEAR DENDRITIC INTEGRATION RULE[*]

WEI P. DAI[†], SONGTING LI[‡], AND DOUGLAS ZHOU[§]

In memory of Professor David Shenou Cai

Abstract. We aim to develop fast algorithms for neuronal simulations to capture the dynamics of a neuron with realistic dendritic morphology. To achieve this, we perform the asymptotic analysis on a cable neuron model with branched dendrites. Using the second-order asymptotic solutions, we derive a bilinear dendritic integration rule to characterize the voltage response at the soma when receiving multiple spatiotemporal synaptic inputs from dendrites, with a dependency on the voltage state of the neuron at input arrival times. Based on the derived bilinear rule, we finally propose two fast algorithms and demonstrate numerically that, in comparison with solving the original cable neuron model numerically, the algorithms can reduce the computational cost of simulation for neuronal dynamics enormously while retaining relatively high accuracy in terms of both sub-threshold dynamics and firing statistics.

Keywords. dendrites; cable equation; asymptotic analysis; dendritic integration; bilinear rule.

AMS subject classifications. 35C20; 92C20.

1. Introduction

Dendrites play an important role in the information processing of a neuron. A neuron in the brain may receive thousands of excitatory (E) and inhibitory (I) synaptic inputs from its dendrites to change the membrane potential at the soma correspondingly under certain rules. This process is known as dendritic integration [20,21]. The dendritic integration of synaptic inputs has been found in experiment to support rich functions of the brain. For instance, the dendritic integration enhances motion detection [10], shapes spiking activity [7,25], and promotes optimal information coding [5] in sensory systems. In addition, the dendritic integration fine tunes brain rhythms, such as modulating the frequency [1] and improving the robustness [31] of gamma oscillations. The dendritic integration also contributes to cognitive computing in the brain [11]. Therefore, to understand the principles of information processing in an individual neuron, as well as in the brain, it is crucial to investigate the computational rules that govern the dendritic integration of synaptic inputs.

In addition to myriads of experiments [26, 27], the rule of dendritic integration is also under active theoretical investigations [14, 17–19, 23]. For instance, it has been shown both in theory and in simulation that the nonlinear integration of synaptic inputs mainly takes place within dendritic branches locally, yielding the computation of logical operations [17]. In addition, a two-layer neuron model and its generalization have been proposed that the synaptic inputs are first integrated nonlinearly within each branch, followed by a linear integration of the branch outputs at the soma [14, 23, 24]. These qualitative models are supported by experiments [2, 15]. However, it remains unclear how to quantitatively capture the dynamics of the membrane potential response at the soma when receiving multiple spatiotemporal synaptic inputs.

[*]Received: January 03, 2019; Accepted (in revised form): July 15, 2019.

[†]School of Physics and Astronomy and Institute of Natural Sciences, Shanghai Jiao Tong University, Shanghai, P.R. China (guess@sjtu.edu.cn).

[‡]Department of Mathematics, MOE-LSC and Institute of Natural Sciences, Shanghai Jiao Tong University, Shanghai, P.R. China (songting@sjtu.edu.cn).

[§]Department of Mathematics, MOE-LSC and Institute of Natural Sciences, Shanghai Jiao Tong University, Shanghai, P.R. China (zdz@sjtu.edu.cn).

An arithmetic rule has been observed in recent experiments that quantitatively characterizes the integration of a pair of E and I inputs [12, 18], which is later referred to as the bilinear dendritic integration rule. The mechanism underlying the bilinear rule has been revealed from the asymptotic analysis of an idealized cable neuron model (a neuron without branched dendrites) in our previous works [18, 19]. Based on the asymptotic analysis, the bilinear rule has been further extended to characterize the integration of multiple synaptic inputs of mixed types, and the rule has been verified in the numerical simulations of a biologically realistic neuron model [18] in the subthreshold dynamical regime (the voltage response is far below the firing threshold). Here, we generalize the previous work by further considering the bilinear rule up to the near-threshold regime of a biologically realistic neuron. By accounting for the fact that the membrane potential of a neuron may strongly fluctuate [30] and even stay at a high voltage level deviating from the resting state [9], we need to consider the dependency of the dendritic integration rule on the state of neuron's membrane potential, whereas the bilinear rule established previously [18] is only derived near a neuron's resting state.

With these generalizations, we then develop fast algorithms for simulation of neuronal dynamics based on an analytically-derived dendritic integration rule from a branched passive cable model with an arbitrary initial state. We evaluate the performance of the algorithms using morphologically realistic neuron simulations. The article is organized as follows. In Section 2, we introduce the mathematical description of a passive neuron with an isotropic soma and branched dendrites. In Section 3, we derive the dendritic integration rule from the asymptotic analysis of the cable neuron model. In Section 4, we introduce two fast algorithms for simulation of neuronal dynamics based on the derived dendritic integration rule, and assess its computational cost and numerical accuracy using a passive cable model of integrate-and-fire type with realistic dendritic morphology. In Section 5, we discuss the source of numerical error and the performance of the algorithms for a neuron with semi-active dendrites.

2. Cable neuron model

Here we consider a passive neuron with an isotropic soma and branched dendrites. The geometry of the soma is modeled as a sphere and that of each dendritic branch is modeled as a cylinder. The topological structure of an example neuron is shown in Figure 2.1A. For the ease of mathematical description, we label dendritic branches with numbers and use local spatial coordinates at each branch. To be specific, we define the coordinate $\vec{x}=(i,x)$ to describe the spatial location x on the ith branch, where $x\in[0,l_i]$ and l_i is the length of the ith branch. We then express the membrane potential of the neuron at location $\vec{x}$ and time t as

$$v(\vec{x},t)=\sum_i v_i(x,t)\chi_i(\vec{x}),$$

where $\chi_i(\vec{x})$ is the characteristic function that equals one if and only if $\vec{x}$ is on the ith branch, and $v_i(x,t)$ describes the neuronal membrane potential at location x on the ith branch. Note that we use the vector coordinate $\vec{x}$ to describe locations on the entire neuron, and we use the scalar coordinate x to describe locations on a specific branch when the branch number, i, is given.

We focus on a small segment $[x,x+\Delta x]$ on the ith branch with diameter d_i. It has shown in experiments that the membrane of the neuron can be viewed as a resistor-capacitor circuit with a constant capacitance and leak conductance density [8, 28]. Therefore, based on the law of current conservation within the segment, the capacitance current of the membrane shall balance with the leak ionic current, the synaptic current across the membrane, and the axial current along the dendrite, i.e.,

$$c\pi d_i \Delta x \frac{\partial v_i}{\partial t} = -g_L \pi d_i \Delta x v_i + I_i^{\mathrm{syn}} + I_i^{\mathrm{a}}(x) - I_i^{\mathrm{a}}(x+\Delta x), \tag{2.1}$$

where v_i is the membrane potential on the ith dendritic branch relative to the resting potential, c is the membrane capacitance density, and g_L is the leak conductance density. Here, I_i^{syn} is the synaptic current on the ith branch

$$I_i^{\mathrm{syn}} = -\sum_{q=E,I} \pi d_i \int_x^{x+\Delta x} G_q \cdot (v_i - \varepsilon_q) dx, \tag{2.2}$$

where G_E and G_I are the excitatory and inhibitory synaptic conductance densities and ε_E and ε_I are their reversal potentials, respectively. When the neuron receives multiple excitatory and inhibitory inputs, G_q is the summation of all individual conductance transients, i.e., $G_q = \sum_j G_{q_j}$ for $q = E,I$. For the jth synaptic input of type q received at time $t = t_{q_j}$ and at site $x = x_{q_j}$, we have $G_{q_j} = f_{q_j} g_{q_j} \delta(x - x_{q_j})$, where f_{q_j} is the input strength of synaptic conductances. The normalized conductance g_{q_j} is often modeled as

$$g_{q_j}(t;t_{q_j}) = N_q(e^{-\frac{t-t_{q_j}}{\sigma_{qd}}} - e^{-\frac{t-t_{q_j}}{\sigma_{qr}}})\Theta(t - t_{q_j}) \tag{2.3}$$

with the peak value normalized to unity by the normalization factor $N_q = [(\frac{\sigma_{qr}}{\sigma_{qd}})^{\frac{\sigma_{qr}}{\sigma_{qd}-\sigma_{qr}}} - (\frac{\sigma_{qr}}{\sigma_{qd}})^{\frac{\sigma_{qd}}{\sigma_{qd}-\sigma_{qr}}}]^{-1}$ and with σ_{qr} and σ_{qd} as the rise and decay time constants [16]. Here $\Theta(t)$ is a Heaviside function. The axial current $I_i^{\mathrm{a}}(x)$ can be derived based on the Ohm's law,

$$I_i^{\mathrm{a}}(x) = -\frac{\pi d_i^2}{4r_i}\frac{\partial v_i}{\partial x}, \tag{2.4}$$

where r_i is the axial resistivity on the ith branch. Taking the limit $\Delta x \to 0$, Equation (2.1) becomes the cable equation describing the spatiotemporal dynamics of the voltage on the ith dendritic branch in response to multiple synaptic inputs,

$$c\frac{\partial v_i}{\partial t} = -g_L v_i - G_E(v_i - \varepsilon_E) - G_I(v_i - \varepsilon_I) + \frac{d_i}{4r_i}\frac{\partial^2 v_i}{\partial x^2}, \tag{2.5}$$

with the conductance $G_q(x,t;\{x_{q_j},t_{q_j}\}) = \sum_{j=1}^{n_q} f_{q_j} g_{q_j}(t;t_{q_j})\delta(x - x_{q_j})$ and n_q is the total number of the synaptic inputs of type q.

We next introduce the boundary conditions of the cable model (Equation (2.5)). For the dendritic branch directly connecting to the soma, e.g., the 1st branch in Figure 2.1A, by applying the law of current conservation at the soma, we have

$$cS\frac{\partial v^s}{\partial t} = -g_L S v^s + I_{dend},$$

where S is the somatic membrane area, v^s is the somatic membrane potential. The dendritic current flowing to the soma, I_{dend}, takes the form of Equation (2.4) at $x = 0$. Because the voltage is continuous at the connection point, *i.e.*, $v^s(t) = v_1(0,t)$, we arrive at the boundary condition on this branch at $x = 0$,

$$c\frac{\partial v_1(0,t)}{\partial t} = -g_L v_1(0,t) + \frac{\pi d_1^2}{4Sr_1}\left.\frac{\partial v_1}{\partial x}\right|_{x=0}. \tag{2.6}$$

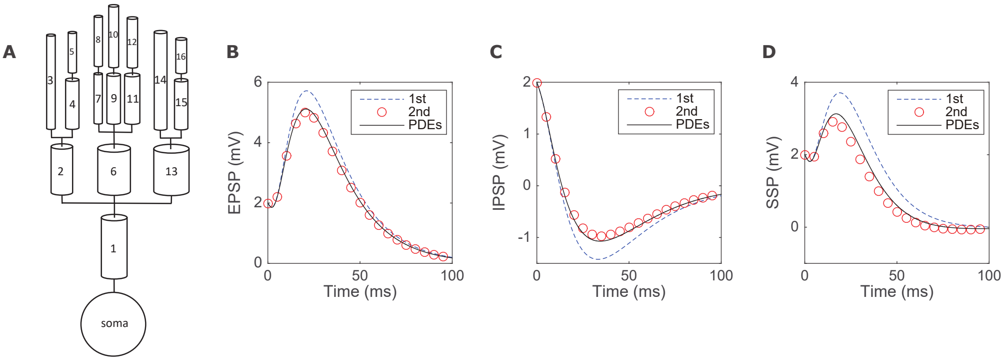

Fig. 2.1. *Example cable neuron models and the asymptotic solutions: (A) Diagram of the branched cable model used in (B-D). (B-D) Asymptotic solutions to various orders to the cable model for (B) EPSP, (C) IPSP and (D) SSP in comparison with numerical solutions to the cable model. The dashed blue line is the first order approximation. The red circle is the second-order approximation. The black solid line is the numerical solution to the PDE cable model in (A). Parameters in our simulation are within the physiological regime [12, 16],* $c=1\mu\mathrm{F}\cdot\mathrm{cm}^{-2}$, $g_L=0.05$ $\mathrm{mS}\cdot\mathrm{cm}^{-2}$, $\varepsilon_E=70$ mV, $\varepsilon_I=-10$ mV, $S=2.8\times10^3$ $\mu\mathrm{m}^2$,$r=100$ $\Omega\cdot\mathrm{cm}$. $\sigma_{Er}=5$ ms, $\sigma_{Ed}=7.8$ ms, $\sigma_{Ir}=6$ ms, $\sigma_{Id}=18$ ms. *The E and I inputs are given at locations* $\vec{x}_E=(9,50$ $\mu\mathrm{m})$ *and* $\vec{x}_I=(6,50$ $\mu\mathrm{m})$, *respectively. The initial voltage of the entire neuron is* 2 mV. *All the voltages are relative to the resting potential.*

For the dendritic branches at the end of the dendrites labeled by $\{i_m\}$, e.g., $i_m\in\{3,5,8,10,12,14,16\}$ in Figure 2.1A, by assuming the ends of these branches are sealed, we arrive at the boundary condition on these branches at $x=l_{i_m}$,

$$\left.\frac{\partial v_{i_m}}{\partial x}\right|_{x=l_{i_m}}=0. \tag{2.7}$$

Additional boundary conditions are prescribed at the connection site of several branches. Suppose that one end of downstream branches labeled by $\{i_n\}$ with diameter $d_{i_1},d_{i_2},\dots,d_{i_k}$ and length $l_{i_1},l_{i_2},\dots,l_{i_k}$, respectively, are connected to the end of the upstream branch i with diameter d_i and length l_i. By the law of current conservation, we have the following boundary conditions,

$$-\frac{\pi d_i^2}{4r_i}\left.\frac{\partial v_i}{\partial x}\right|_{x=l_i}=-\sum_{n=1}^{k}\frac{\pi d_{i_n}^2}{4r_{i_n}}\left.\frac{\partial v_{i_n}}{\partial x}\right|_{x=0}. \tag{2.8}$$

The initial condition of the neuron is set as $v(\vec{x},0)=v_0$, which can deviate from its resting state. This is consistent with experimental observations that the membrane potential of a neuron in general fluctuates [30] and may stay at a high voltage level deviating from the resting state [9] before receiving any future synaptic inputs.

3. Derivation of the dendritic integration rule

In this section, we first derive the dendritic integration rule for a pair of E and I synaptic inputs, and then generalize the rule to characterize the dendritic integration of multiple synaptic inputs of mixed types.

In the absence of synaptic inputs, the cable equation (Equation (2.5)) with the boundary conditions (Equations (2.6-2.8)) is a linear system. Therefore, its Green's function in response to a delta-pulse input exists. However, because the synaptic currents (Equation (2.2)) include the unknown dynamical variable v_i, the Green's function method cannot be directly applied to obtain the analytical solution to Equation (2.5) when the synaptic inputs are present. Note that, when the neuron receives a pair of E and I inputs, within the physiological regime, i.e., the amplitude of an excitatory postsynaptic potential (EPSP) being less than 5 mV and the amplitude of an inhibitory postsynaptic potential (IPSP) being less than 2 mV, the corresponding input strengths f_E and f_I are relatively small. Therefore, given an E input at $\vec{x}_E=(i_E,x_E)$ and an I input at $\vec{x}_I=(i_I,x_I)$, we can represent the voltage response $v(\vec{x},t)$ as an asymptotic series in the powers of f_E and f_I,

$$v(\vec{x},t;\mathcal{W})=\sum_{k=0}^{\infty}\sum_{m+n=k} f_E^m f_I^n v_{mn}(\vec{x},t;\mathcal{W}), \tag{3.1}$$

where $\mathcal{W}\subseteq\{\vec{x}_E,\vec{x}_I,t_E,t_I\}$ is parameter space, $\vec{x}_E,t_E\in\mathcal{W}$ if $m\neq 0$; $\vec{x}_I,t_I\in\mathcal{W}$ if $n\neq 0$. In addition, for each branch i, we define

$$v_{i_{mn}}(x,t;\mathcal{W})=v_{mn}(\vec{x},t;\mathcal{W})\chi_i(\vec{x}). \tag{3.2}$$

Combining Equations (3.1-3.2) and the cable equation (Equation (2.5)), order by order, we can obtain a differential equation for each branch i. For the zeroth-order, we have

$$c\frac{\partial v_{i_{00}}}{\partial t}=-g_L v_{i_{00}}+\frac{d_i}{4r_i}\frac{\partial^2 v_{i_{00}}}{\partial x^2}.$$

Using the boundary conditions (Equations (2.6-2.8)) and the initial condition $v(\vec{x},t)=v_0$, the solution is simply

$$v_{00}(\vec{x},t;\varnothing)=v_0 e^{-\frac{t}{\tau_m}} \tag{3.3}$$

for $\vec{x}$ on any branch of the neuron. Here $\tau_m=c/g_L$ is the membrane time constant. For the first order of excitation $O(f_E)$,

$$c\frac{\partial v_{i_{10}}}{\partial t}=-g_L v_{i_{10}}+\frac{d_i}{4r_i}\frac{\partial^2 v_{i_{10}}}{\partial x^2}-g_E(t;t_E)\delta(\vec{x}-\vec{x}_E)(v_{00}-\varepsilon_E).$$

With the help of Green's function, the solution can be expressed as

$$v_{10}(\vec{x},t;\vec{x}_E,t_E)=G(\vec{x},\vec{x}_E,t)*[g_E(t;t_E)(\varepsilon_E-v_{00}(\vec{x}_E,t;\varnothing))]. \tag{3.4}$$

Here '$*$' denotes convolution in time, $G(\vec{x},\vec{y},t)$ is the Green's function of the system. For the second-order of excitation $O(f_E^2)$,

$$c\frac{\partial v_{i_{20}}}{\partial t}=-g_L v_{i_{20}}+\frac{d_i}{4r_i}\frac{\partial^2 v_{i_{20}}}{\partial x^2}-g_E(t;t_E)\delta(\vec{x}-\vec{x}_E)v_{10}. \tag{3.5}$$

Because v_{10} is given by Equation (3.4), the solution of Equation (3.5) is

$$v_{20}(\vec{x},t;\vec{x}_E,t_E)=G(\vec{x},\vec{x}_E,t)*[-g_E(t;t_E)v_{10}(\vec{x}_E,t;\vec{x}_E,t_E)].$$

The above procedure can be generalized to higher orders. Similarly, we can have the first and second-order solutions to the inhibitory inputs.

$$v_{01}(\vec{x},t;\vec{x}_I,t_I)=G(\vec{x},\vec{x}_I,t)*[g_I(t;t_I)(\varepsilon_I-v_{00}(\vec{x}_I,t;\varnothing))], \tag{3.6}$$

$$v_{02}(\vec{x},t;\vec{x}_I,t_I)=G(\vec{x},\vec{x}_I,t)*[-g_I(t;t_I)v_{01}(\vec{x}_I,t;\vec{x}_I,t_I)].$$

For the order of $O(f_Ef_I)$, we have

$$c\frac{\partial v_{i_{11}}}{\partial t}=-g_Lv_{i_{11}}+\frac{d_i}{4r_i}\frac{\partial^2 v_{i_{11}}}{\partial x^2}-g_E(t)\delta(\vec{x}-\vec{x}_E)v_{01}-g_I(t)\delta(\vec{x}-\vec{x}_I)v_{10},$$

whose solution is obtained as follows,

$$\begin{aligned} v_{11}(\vec{x},t;\vec{x}_E,\vec{x}_I,t_E,t_I)=&G(\vec{x},\vec{x}_E,t)*[-g_E(t;t_E)v_{01}(\vec{x}_E,t;\vec{x}_I,t_I)]\\ &+G(\vec{x},\vec{x}_I,t)*[-g_I(t;t_I)v_{10}(\vec{x}_I,t;\vec{x}_E,t_E)]. \end{aligned} \tag{3.7}$$

All these asymptotic solutions involve the Green's function of the system. Although the Green's function of cable models in general is difficult to obtain analytically, it can be measured numerically using white-noise input as a property of linear systems. The numerically measured Green's function enables one to directly evaluate the performance of the asymptotic solutions in approximating the numerical solutions to the cable model. When the cable neuron model (Figure 2.1A) receives a pair of E and I synaptic inputs, our numerical simulation of the model using the Crank-Nicolson method (time step 0.01 ms and space step 1 μm) shows that the second-order asymptotics is sufficiently accurate to capture the solution of physiological membrane potentials, as demonstrated in Figure 2.1B-D. Therefore, the EPSP denoted by V_E induced by an individual E input alone can be approximated by $V_E\approx v_{00}+f_Ev_{10}+f_E^2v_{20}$, the IPSP denoted by V_I induced by an individual I input alone can be approximated by $V_I\approx v_{00}+f_Iv_{01}+f_I^2v_{02}$, and the summed somatic potential (SSP) denoted by V_S induced by both E and I inputs can be approximated by $V_S\approx v_{00}+f_Ev_{10}+f_E^2v_{20}+f_Iv_{01}+f_I^2v_{02}+f_Ef_Iv_{11}$.

If we normalize these voltages by subtracting the zeroth order, we can identify a *bilinear dendritic integration rule* to capture the membrane potential response when the neuron receives a pair of E and I inputs, i.e.,

$$\widehat{V}_S=\widehat{V}_E+\widehat{V}_I+k_{EI}\widehat{V}_E\widehat{V}_I, \tag{3.8}$$

where $\widehat{V}_p=V_p-v_{00}$ for $p\in\{E,I,S\}$, and the coefficient k is defined as

$$k_{EI}(t;\vec{x}_E,\vec{x}_I,t_E,t_I)\equiv\frac{\widehat{V}_S-\widehat{V}_E-\widehat{V}_I}{\widehat{V}_E\cdot\widehat{V}_I}=\frac{v_{11}(t;\vec{x}_E,\vec{x}_I,t_E,t_I)}{v_{10}(t;\vec{x}_E,t_E)\cdot v_{01}(t;\vec{x}_I,t_I)}+\mathcal{O}(f_E)+\mathcal{O}(f_I). \tag{3.9}$$

Equation (3.9) shows that, for a fixed initial condition $v(\vec{x},t)=v_0$, the leading order of k_{EI} as a function of time is independent of the input strength. Note that the variable $\vec{x}$ is omitted in the expression of k_{EI} and v_{mn} because we are only concerned with the voltage response at the soma. In addition, Equations (3.4, 3.6, 3.7) indicate that k_{EI} is determined by the input locations and input arrival times. Therefore, k_{EI} parameterizes the spatiotemporal information of the synaptic inputs. Note that the bilinear integration rule (Equation (3.8)) holds at any location on the neuron, including its soma in particular.

Based on a similar analysis, it is straightforward to generalize Equation (3.8) to describe the dendritic integration of multiple synaptic inputs of mixed types,

$$\widehat{V}_S=\sum_i\widehat{V}_E^i+\sum_j\widehat{V}_I^j+\sum_{m,n}k_{EI}^{mn}\widehat{V}_E^m\widehat{V}_I^n+\sum_{u,w}k_{EE}^{uw}\widehat{V}_E^u\widehat{V}_E^w+\sum_{r,s}k_{II}^{rs}\widehat{V}_I^r\widehat{V}_I^s, \tag{3.10}$$

where $\widehat{V}_E^i$ and $\widehat{V}_I^j$ are the normalized individual EPSP and IPSP, respectively, k_{EI}^{mn}, k_{EE}^{uw}, and k_{II}^{rs} are the coefficients encoding the spatiotemporal information of the synaptic inputs, which are nearly independent of synaptic input strengths.

The derived bilinear integration rule provides an alternative approach to simulate the dynamics of a spatial neuron rather than numerically solving the PDE cable models. By taking advantage of its simple form, next we will develop fast simulation algorithms and evaluate their performance including computational speed and numerical accuracy in contrast to the full simulation of a morphologically realistic passive neuron and the simulation based on the linear integration rule below (without taking into account the bilinear interaction),

$$\widehat{V}_S = \sum_i \widehat{V}_E^i + \sum_j \widehat{V}_I^j. \tag{3.11}$$

4. Fast algorithms based on the dendritic integration rule

Here we develop two fast algorithms based on the derived bilinear dendritic integration rule and assess its performance using a passive cable neuron model with full dendrites. The neuron is passive in a sense that there is no active ion channel on the soma or any dendrites. The morphology of the example neuron is reconstructed from a hippocampal CA1 pyramidal neuron (data acquired from the Duke Southampton Archive of neuronal morphology [4]), and is shown in Figure 4.1A. The time of the onset of an action potential (spike time) in the passive neuron is determined by the time when the membrane potential of the soma crosses a threshold $V_T = -55$ mV. When the neuron spikes, we reset $v(\vec{x},t)$ across the neuron uniformly to its resting potential $V_r = -70$ mV. A case for a semi-active neuron that is capable of generating a full-spike shape (with the same algorithms) is shown in Section 5.

We develop fast algorithms by employing the bilinear dendritic integration rule together with a library approach to obtain the voltage trace at the soma in response to multiple synaptic inputs. The voltage dependence of the bilinear integrations is the initial voltage of the second input of each pair. To evaluate the fast algorithms, we then evolve the PDE cable model numerically by using the NEURON software [13] as the numerical solver to obtain the neuronal voltage trace under the same input condition as the benchmark for comparing the computational speed of the two approaches, as well as their numerical accuracy in terms of voltage trace, spike timing, and other statistics such as the gain curve and the distribution of inter-spike intervals. The NEURON software is also used for the setup of the library. The source code of the fast algorithms is available on *https://github.com/g13/bNEURON.*

4.1. Library setup.

For the purpose of illustration, and without losing generality, we randomly choose six excitatory synapses that are located far away from the soma, and three inhibitory synapses located relatively near the soma. The spatial distribution of excitatory and inhibitory synapses in our setup is consistent with that observed in experiment for CA1 pyramidal neurons [22]. At each synaptic input location, we use four different magnitudes of input strength $f \in \{0.2, 0.4, 0.8, 1.6\}$ nS for the synaptic conductance. The largest input strength will induce a voltage change at the soma as high as 4 mV when $v_0 = -62$ mV, as shown in Figure 4.1B and C. For better understanding, the various time-related variables used in the library (and later in the fast algorithms) are schematically explained in Figure 4.2A.

4.1.1. Library of single-input voltage trace.

We first build a library of a series of somatic voltage responses $\widehat{V}_p(t; f, v_0, \Delta t)$ induced by individual inputs at

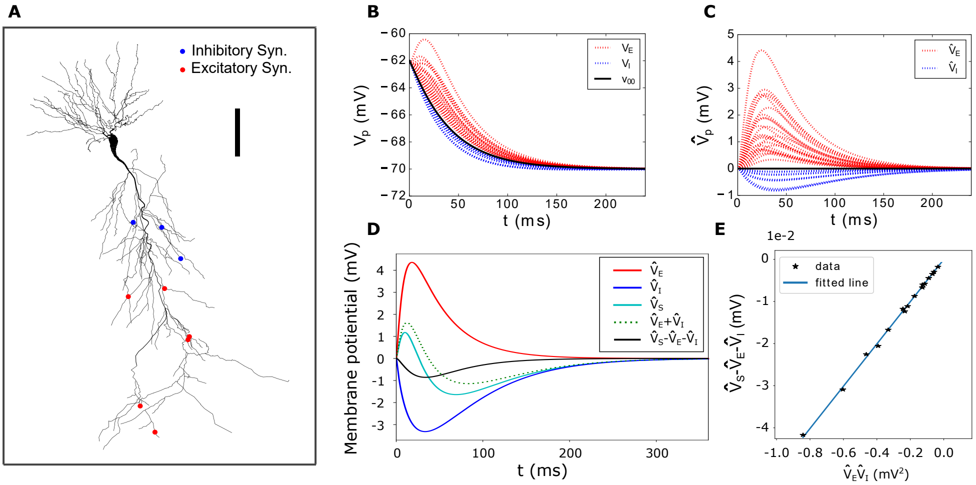

Fig. 4.1. *Library setup: (A) The dendritic morphology of the passive neuron for evaluating the fast algorithms we developed. The six red dots are the excitatory synapses and the three blue dots are the inhibitory synapses. (B) The responses of membrane potential $V_p(t)$ at the soma to each single synaptic input at the dendritic positions marked in (A). The black line marks V_{base} when no input is presented. Each synapse is associated with four different input strengths, all plotted with $v_0 = -62$* mV*. (C) Voltage traces of $\widehat{V}_p(t) = V_p(t) - V_{base}$, which are the single-input response at the soma to be stored in the library. Only the case of $\Delta t = 0$* ms *is shown here (see text for the definition of Δt). (D) One example plot of the contributions from single-input response to a two-input case ($\Delta t = 0$). The response $\widehat{V}_E$ to one excitatory input is plotted in red, and the response $\widehat{V}_I$ to one inhibitory input is plotted in blue. The full response of the soma $\widehat{V}_S$ when receiving the two inputs simultaneously is plotted in cyan, and the linear sum of the two single-input response $\widehat{V}_E + \widehat{V}_I$ is plotted in dotted green. The residual term $\widehat{V}_S - \widehat{V}_E - \widehat{V}_I$ for the calculation of the second-order coefficient k (Equation (3.9)) is plotted in black. (E) Linear relation between the residual term versus the product of the two single-input response $\widehat{V}_E\widehat{V}_I$. For each combination (a total of 16) of such pairs of synaptic inputs in (D), we acquire a data point, and we fit a line through these data points when $|\widehat{V}_E\widehat{V}_I|$ reaches its maximum at $t = 39.9$* ms *with $v_0 = -66$* mV*. In this example, the slope $k = 0.05$* mV^{-1}.

various synaptic sites, which we later refer to as the “single-input responses”. The subscript p here denotes the numbering index of the synapses and $t \in [0, D_p]$, where D_p is the duration when $\widehat{V}_p$ decays back to sufficiently close to zero. A typical value of D for excitatory synaptic inputs is 240 ms, and 300 ms for inhibitory ones. The input strength is denoted by f, and the initial voltage at the soma is denoted by v_0. We further evaluate the single-input responses for different values of Δt, which denotes the duration of a voltage clamp at v_0, starting from $t = 0$ when the input arrives. The meaning of Δt differs when the single-input responses are used in different situations. For the generation of the library of the second-order coefficient k in the next subsection, where the single-input responses are also needed; then, for the second synaptic input of the pair, Δt means the time interval between the two inputs since the initial voltage dependence is defined by the initial voltage at the arrival time of the second synaptic input. When a spike is generated before $t = D_p$, linear contributions from the single-input responses need to be re-evaluated after the spike since the voltage is reset to the resting potential; then, in this case, Δt means the time difference between the synaptic input and the time of spike, illustrated as $\Delta t'$ in Figure 4.2A. When used for linear contributions to the voltage trace before a spike, Δt simply equals zero.

To obtain $\widehat{V}_p(t)$ with an initial voltage of v_0, we first perform the simulation without

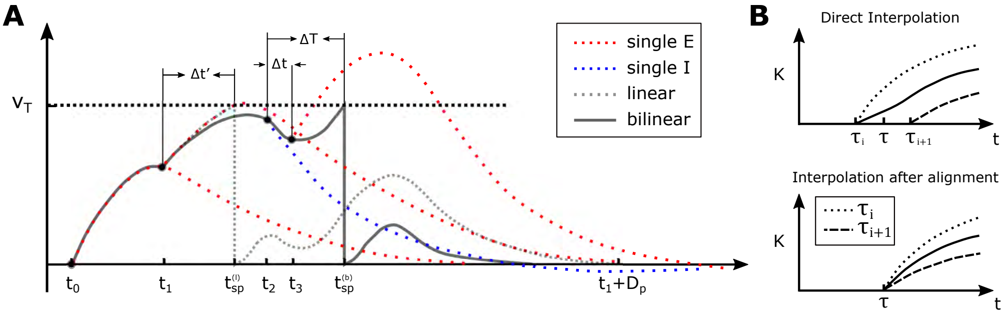

FIG. 4.2. *Schematic diagrams: The time-related variables Δt, ΔT, and D_p that considered in the library setup of single-input voltage trace and second-order coefficient are illustrated in (A) and interpolation procedure in (B). (A) t_0,t_1,t_2, and t_3 are the arrival time of the four synaptic inputs. Assuming the input received at t_1 is from the pth synapse, D_p denotes its duration of contribution and $\Delta t'$ marks the time difference between the spike time of linear integration (dotted grey) $t_{sp}^{(l)}$, and synaptic input time t_1. The time difference between the two synaptic inputs t_2 and t_3 is denoted by Δt; the time difference between the spike time of the bilinear integration (grey) $t_{sp}^{(b)}$ and t_2 is denoted by ΔT. The initial voltage for the single-input responses ($\widehat{V}_p(t)$), excitatory in red and inhibitory in blue, are based on the voltage trace of the bilinear integration. Note that the action potential is generated much earlier without taking into account the bilinear integration terms. After the neuron spikes at $t_{sp}^{(l)}$ ($t_{sp}^{(b)}$), the linear (bilinear) integration procedures are to be repeated for all inputs (input pairs) that come before the spike with new variables of interpolation, e.g, $\Delta t'$ for the single-input responses arrived at t_1, and ΔT for second-order coefficients of the pair of inputs arrived at t_2 and t_3, respectively. (B) The top figure illustrates the situation of a direct interpolation on $\Delta t = \tau$ for second-order coefficient k (see Section 4.1.3) while the bottom figure shows the correct way: align the two nearest library data τ_i and τ_{i+1} to τ to avoid undefined nonzero values for $t \in [\tau_i, \tau]$ shown in the top figure. The solid line in both figures are the supposedly interpolated k with the two methods, respectively. Note that, (B) can also be viewed as the alignment of Δt for single-input responses.*

any input to record $V_{base}(t)$ (corresponding to the term $v_{00}(t)$ in Equation (3.3)), which is the zeroth-order response. Then, the simulation is performed when the neuron receives individual inputs from various synaptic sites, with all the combinations of input strength f and discretized values of Δt to obtain a series of $V_p(t)$. The resultant $V_p(t)$ are either EPSP traces above $V_{base}(t)$, or IPSP traces that go below it, where $V_p(0) = v_0$ and $V_{base}(0) = v_0$. An example with $v_0 = -62$ mV and $\Delta t = 0$ ms is shown in Figure 4.1B, where $V_{base}(t)$ is the black line, excitatory traces of $V_p(t)$ are in red dotted lines and inhibitory ones are in blue dotted lines. Note that, all the combinations of input strengths and synaptic locations are plotted. Then, by subtracting $V_{base}(t)$ from $V_p(t)$, we obtain $\widehat{V}_p(t)$ in Figure 4.1C for each combination of the discretized values of f and Δt at the pth synaptic location for the given initial voltage v_0. Finally, we iterate the process over the discretized values of v_0 to obtain the full library of $\widehat{V}_p(t; f, v_0, \Delta t)$.

4.1.2. Library of the second-order coefficient k. Next we build the library for the second-order coefficient $k(t; v_0, \vec{x}_p, \vec{x}_q, \Delta t)$, where $\Delta t = t_q - t_p$ marks the interval between the time of input of the pth synapse and that of the qth synapse. Because the locations of synaptic inputs are now fixed, we change $\vec{x}_p$ and $\vec{x}_q$ to be the subscripts of k, and rewrite the coefficient as $k_{pq}(t; v_0, \Delta t)$ for the sake of simplicity. Without losing generality, we assume that the input from the pth synapse comes before the input from the qth synapse. To measure the second-order coefficient k, in addition to the single-input response measured previously, we need to obtain the two-input full response V_S at the soma based on Equation (3.9). The first input from the pth synapse

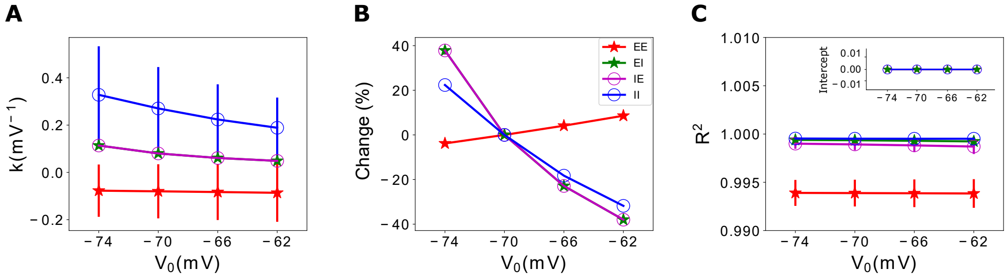

Fig. 4.3. *Validation of the two-input integration rule: (A) Plots of the second-order coefficient k for all pairs of synaptic inputs as a function of v_0, where the pairs are categorized into combinations of EE, EI, IE, and II. The errorbars give the standard deviations for different combinations of synaptic locations. The coefficient is evaluated for each pair of synapses at the maximum of the product of their responses $|\widehat{V}_p\widehat{V}_q|$ when receiving the two synaptic inputs simultaneously, $\Delta t = 0$. (B) Plots of the percentage of change in k, with respect to the k at $v_0 = V_r$, across different v_0 for data in (A). The errorbars are too small to be visible. (C) Plots of the goodness of fit R^2 at different v_0. The errorbars show the mean and averaged standard deviation of* $\Delta t \in \{0,4,8,12\}$ ms, $t \in \{4,8,12,20,30,50,70,130\}$ ms *and all combinations of pairs of synapses. The inset figure shows the intercept of the fitted lines.*

arrives at $t_p = 0$ and the second input from the qth synapse arrives at $t_q = \Delta t$. Just as the generation of single-input response $V_p(t)$ for $\Delta t > 0$ in the previous section, we need to clamp the voltage at v_0 for a duration of Δt. Note that although the voltage is clamped at v_0, the evolution of conductance from the first input is not disrupted. Also note that Δt only needs to be discretized over $[0, D_p]$, the contribution of bilinear integration terminates as the first input contribution ends, which is originated from the pth synapse. After subtracting $V_{base}(t)$ from the summed response V_S, we obtain a series of $\widehat{V}_S$ corresponding to the pair of inputs from the pth and qth synapses for all combinations of input strengths f, respectively, with different values of discretized v_0 and Δt. Finally, for $t \in (\Delta t, D_p]$ and initial voltage v_0, we can identify a linear relation between the dataset of $\widehat{V}_S - \widehat{V}_p - \widehat{V}_q$ and $\widehat{V}_p\widehat{V}_q$, where the slope of the linear fitting is a sound approximation of the second-order coefficient k at time t if the intercept is zero. The fitting process for one of the pairs is shown in Figure 4.1D and E as an example. Note that before the arrival of the second input, i.e., $t \in [0, \Delta t]$, we set k_{pq} to be zero because the bilinear interaction between the two inputs has not happened yet.

Figure 4.3A shows the typical values of second-order coefficient k for different initial voltages v_0 and synaptic input pairs from different locations and with different input strengths. From the negligible errorbars for k_{EI} or k_{IE}, we confirm that with the setup of inhibitory synapses being closer to the soma than the excitatory synapses, the coefficients k_{EI} and k_{IE} are nearly independent of the location of the excitatory synapse, as studied in [12, 18]. However, the coefficient experiences a ∼60% change across the range of v_0, as shown in Figure 4.3B. It is easily recognized that as v_0 increases, k_{EI}, k_{IE}, and k_{II} are monotonically decreasing for pairs of synaptic inputs at different locations. However k_{EE} only changes a little for different v_0, which can be explained by the fact that the change of the membrane potential by the initial voltage v_0 within the dynamic range of sub-threshold membrane potential is small compared to the reversal potential of excitatory inputs. Figure 4.3C shows that the goodness of fit R^2 across v_0 only deviates slightly from unity, and the errorbars representing the standard deviation of all combinations of parameters are negligible. The inset figure shows that the intercept of the line from linear fitting is always close to zero independent of v_0, t, Δt, or synaptic location. Thus, the result confirms the bilinear integration rule (Equation (3.8)) and that the second-order coefficient is indeed independent of input strength within the whole parameter space for the passive neuron; yet, it is dependent on the initial state of the neuron's voltage distribution and the spatiotemporal information of the synaptic inputs.

An extended library for second-order coefficient that involves spiking events can be built by taking care of the discontinuity arising from the spike-reset dynamics in the passive neuron. The voltage across the neuron is reset to the resting potential once reaching the spiking threshold, this rapid change in the voltage distribution erases the dependence of bilinear integration on the initial voltage, while the evolution of conductance is unaffected. Then, the interpolation has to be redone after the neuron reset for the pairs of inputs that are received before the spike. Thus, a new dimension must be created when building the library, just for the second-order coefficient k. To be specific, we add to k another dependence on ΔT (Figure 4.2A) that denotes the interval between the spike-reset time and the arrival time of the first input, which is always greater or equal to the time interval between the two inputs Δt.

4.1.3. Interpolation of the library. Finally, to apply the library approach in the fast algorithms, one needs to first define the interpolation procedure for the library. The interpolation shall avoid having nonzero single-input voltage responses $\widehat{V}_p(t)$, and nonzero second-order coefficient $k_{pq}(t)$ before the second input, from the qth synapse, arrives $(t<t_q)$. To achieve this, one needs to first align and interpolate the library data in the dimension of time intervals, e.g. Δt and ΔT, before interpolating along other dimensions. For the sake of illustration, we take the interpolation of $k_{pq}(t;v_0,\Delta t)$ along the dimension of v_0 as an example. To interpolate along the dimension of v_0, one needs to first align and interpolate along the dimension of Δt. Let τ_i and τ_{i+1} $(\tau_{i+1}>\tau_i)$ denote the two nearest discretized values to $\Delta t=\tau$, and $\tilde{v}_m$ $(m=j,j+1)$, denotes the two nearest discretized values to $v_0=\tilde{v}$ in the library. We align the library data of $k_{pq}(t;\tilde{v}_m,\tau_i)$, $t>\tau_i$ with the library data of $k_{pq}(t;\tilde{v}_m,\tau_{i+1})$, $t>\tau_{i+1}$ for $m=j,j+1$ to avoid nonzero values caused by $t\in[\tau_i,\tau]$ in the library data of $k_{pq}(t;\tilde{v}_m,\tau_i)$ if interpolated directly, as illustrated in Figure 4.2B. Now one can interpolate over τ to get $k_{pq}(t;\tilde{v}_j,\Delta t)$ and $k_{pq}(t;\tilde{v}_{j+1},\Delta t)$, and finally perform the interpolation for $\tilde{v}\in[\tilde{v}_m,\tilde{v}_{m+1}]$.

4.2. The fast algorithms. Once we have built the library, we can speed up enormously the simulation of a passive neuron with full dendritic morphology by implementing the arithmetic dendritic integration scheme that follows Equation (3.10). Because each single-input response $\widehat{V}_p$ will decay to zero after the decay time scale D_p, we only need to sum over the input pairs that have input time difference Δt smaller than the decay time scale D_p for a *bilinear integration* scheme. Two bilinear integration schemes are introduced in the following section, the *full voltage trace* scheme and the *partial voltage trace* scheme, with the latter one focusing more on computational efficiency. The *linear integration* scheme is simply the *full voltage trace* scheme without the bilinear contributions. The subscripts i, j, l from here on denote the sequence of synaptic inputs in time, and are independent of p, q that denote both the synaptic type and location.

4.2.1. Full voltage trace scheme. To efficiently generate the entire voltage trace in response to multiple inputs, we develop a *full voltage trace* scheme, where for each synaptic input received at t_i we calculate its linear contribution and all the bilinear contribution originated from the interactions with previous inputs to $v(t)$ for $t>t_i$ forward in time. We provide the pseudo code for the scheme in Alg. 1, and an overview of the scheme as the following.

Algorithm 1: Full voltage trace scheme

```
build and load library
$n \leftarrow$ the total number of inputs
add $V_{base}(t)$ to $v(t)$
for $i \leftarrow 1$ to $n$ do
    /* $p$ denotes the synaptic index of the $i$th input */
    interpolate to find $\widehat{V}_p$ at $f = f_i$, $v_0 = v(t_i)$ and $\Delta t = 0$
    add $\widehat{V}_p$ to $v(t), t \in [t_i, t_i + D_p]$
    $j \leftarrow i - 1$ // $q$ denotes the synaptic index of the $l$th input
    while $j > 0$ and $t_j + D_q > t_i$ do
        interpolate to find $k_{qp}$ at $v_0 = v(t_i)$ and $\Delta t = t_i - t_j$
        add $k_{qp}\widehat{V}_q\widehat{V}_p$ to $v(t), t \in [t_i, t_j + D_q]$
        $j \leftarrow j - 1$
    end
    /* the following deals with spike-reset */
    while $v(t) > V_T, t \in [t_i, t_i + D_p]$ do
        $v(t) \leftarrow V_r, t \in [t_i, t_i + D_p]$
        $t_{sp} \leftarrow$ time of $v(t)$ cross $V_T$
        $j \leftarrow i$ // $p$ denotes the synaptic index of the $i$th input
        while $j > 0$ and $t_j + D_p > t_{sp}$ do
            interpolate to find $\widehat{V}_p$ at $f = f_p$, $v_0 = V_r$ and $\Delta t = t_{sp} - t_j$
            add $\widehat{V}_p$ to $v(t), t \in [t_{sp}, t_j + D_p]$
            $l \leftarrow j + 1$ // $q$ denotes the synaptic index of the $l$th input
            while $l < i$ do
                interpolate to find $k_{pq}$ at $v_0 = V_r$, $\Delta t = t_l - t_j$ and $\Delta T = t_{sp} - t_j$
                add $k_{pq}\widehat{V}_p\widehat{V}_q$ to $v(t), t \in [t_{sp}, t_j + D_p]$
                $l \leftarrow l + 1$
            end
            $j \leftarrow j - 1$
        end
    end
end
```

For the ith input received from the pth synapse at t_i, we obtain its single-input response $\widehat{V}_p(t'; v_0 = v(t_i), \Delta t = 0)$ by interpolation from the library for $t' \in [0, D_p]$ and add it to the membrane potential $v(t)$ at the soma for $t \in [t_i, t_i + D_p]$. Then for the jth input received from the qth synapse at t_j before the ith input arrival time t_i and satisfies $t_j + D_q > t_i$, we interpolate the jth single-input response and add to $v(t)$ (for $t \in [t_i, t_j + D_q]$) the bilinear contribution from the input pair (j, i), which can be written into

$$k_{qp}(t'; v(t_i), t_i - t_j)\widehat{V}_q(t'; v(t_i), t_i - t_j)\widehat{V}_p(t' - (t_i - t_j); v(t_i), 0), \tag{4.1}$$

where $t' \in [t_i - t_j, D_q]$ is relative to the arrival time of the jth synaptic input, t_j. We repeat the process for all the inputs satisfying the above condition. When $v(t)$ crosses the threshold V_T at $t = t_{sp}$, an action potential is considered to be generated and we reset the $v(t)$ across the neuron back to V_r. Before continuing with the $(i+1)$th input

Algorithm 2: Partial voltage trace scheme

```
    build and load library
    $n \leftarrow$ the total number of inputs
    for $i \leftarrow 1$ to $n$ do
        update the following variables for interpolation (VI) of the $i$th input:
        $v_0 = v(t_i)$, $\Delta t = 0$, $f = f_i$
1       add $V_{base}(t_i)$ to $v(t_i)$
        $j \leftarrow i-1$ // $p$ denotes the synaptic index of the $j$th input
2       while $j > 0$ and $t_j + D_p > t_i$ do
            interpolate to find $\widehat{V}_p$ for the $j$th input
            add $\widehat{V}_p(t_i - t_j)$ to $v(t_i)$
            $l \leftarrow j+1$
            update the VI of input pair $(j,i)$:
            $v_0 = v(t_j)$, $\Delta t = t_i - t_j$
            while $l < i$ do
                /* $q$ denotes the synaptic index of the $l$th input */
                interpolate to find $k_{pq}$ with the VI for the input pair $(j,l)$
                add $k_{pq}\widehat{V}_p\widehat{V}_q$ to $v(t_i)$
                $l \leftarrow l+1$
            end
            $j \leftarrow j-1$
        end
        /* The following deals with spike-reset */
        if $v(t_i) > V_T$ then
            use linear interpolation over $v(t)$ to find the time of spike $t_{sp}$
            /* repeat line 1 to line 2 (skip updating VI) when evaluate
                $v(t)$, $t \in [t_{i-1}, t_i]$ for interpolation of $t_{sp}$ */
            for each input $j$: $j < i$ and $t_j + D_p \geq t_i$ do
                /* $p$ denotes the synaptic index of the $j$th input */
                update the VI of the $j$th input:
                $v_0 = V_r$, $\Delta t = t_{sp} - t_j$ // $f$ is unchanged
                for each input $l$: $l > j$ and $l < i$ do
                    update the VI for the input pair $(j,l)$:
                    $v_0 = V_r$ // $\Delta t$ is unchanged
                    additional dependence of $\Delta T = t_{sp} - t_j$
                end
            end
            $i \leftarrow i-1$ // rerun the current iteration with the new VIs
        end
    end
```

through the same process above, we need to first update $v(t)$ for $t \in [t_{sp}, t_i + D_p]$, which means to reapply the single-input responses and bilinear contributions to $v(t)$ but with a new set of parameters due to the spike-reset of the voltage. Specifically, for each input received at time t_j we need to interpolate the jth single-input response from the qth synapse with $v_0 = V_r$ and $\Delta t = t_{sp} - t_j$ (e.g., $\Delta t'$ in Figure 4.2A). For each input pair (j,l), $t_l > t_j$, we need to interpolate the second-order coefficient k with $\Delta t = t_l - t_j$ and an additional dependence on $\Delta T = t_{sp} - t_j$ from the extended library of k, e.g, see the inputs arrived t_2 and t_3 in Figure 4.2A. When the second-order term is dropped in this algorithm, we call the scheme a *linear integration* scheme.

4.2.2. Partial voltage trace scheme. If the detailed voltage trace is not of interest, then we only need to update the membrane potential at the arriving time of each input to evolve the voltage. To achieve this, we can introduce a *partial voltage trace* scheme. The pseudo code for the scheme is provided in Alg. 2. We summarize here the most important step in this scheme: At the arriving time $t = t_i$ of the ith input, we look backwards in time to find inputs that arrive at $t = t_j < t_i$ from the pth synapse that satisfies $t_i - t_j < D_p$ for all j and p; and find the input pairs (j, l) that satisfy $t_j < t_l < t_i$. We then add their linear and bilinear contributions, respectively, to the somatic voltage only at this time instant $v(t = t_i)$, as opposed to the *full voltage trace* scheme introduced previously.

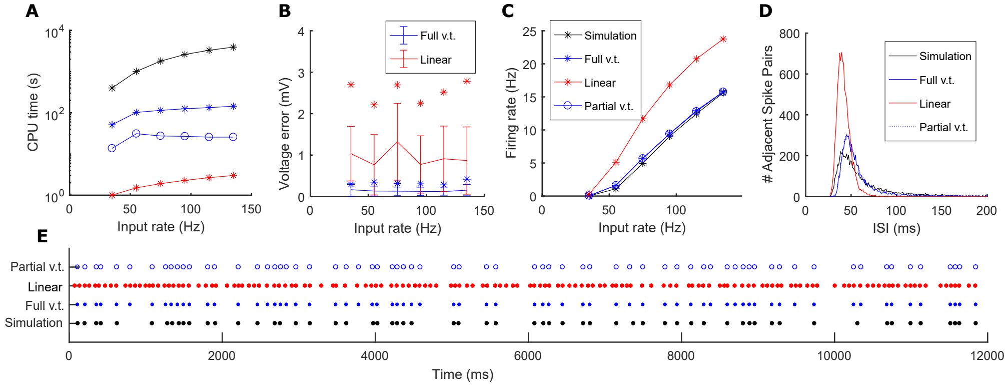

FIG. 4.4. *The performance of the fast algorithms for a passive neuron with full dendrites shown in Figure 4.1A: Statistics are plotted from four trials of 124 seconds for six different Poisson input rates, the synaptic inputs are generated from independent Poisson processes for each synapse. (A) Computational costs, in terms of CPU time spent, against input rate for the four schemes, i.e., the simulation (NEURON software) in black, full voltage trace (full v.t.) scheme in blue asterisks, partial voltage trace (partial v.t.) scheme in blue circles, and linear integration scheme in red asterisks. Notice that the ordinate is in log scale. (B) The mean and standard deviations of the difference between the voltage traces of the simulation and those of the two schemes (errorbars), i.e., linear integration scheme in red, and full voltage trace scheme in blue, per time step before the first spike. The accumulated errors at the last time step before the first spike (asterisks). Note that the partial voltage trace scheme is not included since it does not have a complete voltage trace. (C) Gain curves for the four different schemes. (D) The distribution of inter-spike intervals for each scheme, collected from four trials with Poisson input rates of* 135 Hz. *(E) An excerpt of raster plot of the spike timings, with the same Poisson input rate of* 95 Hz *for each scheme.*

4.2.3. Performance of the fast algorithms. The arithmetic schemes demonstrate a massive (hundred-fold) reduction of computational cost compared with a direct numerical computation of the PDE cable neuron model solved by the NEURON software, as shown in Figure 4.4A. In practice, the library can be stored in the memory contiguously and a modern CPU can parallelize the extra additions and multiplications to a certain extent. Thus, the efficiency gained from replacing *full voltage trace* scheme with the *partial voltage trace* scheme is less significant, compared to the computational time saved by dropping the numerical simulation of the PDE cable neuron model performed by the NEURON software.

Meanwhile, the relative accuracy of the bilinear integration schemes, compared with the *linear integration* scheme, are retained in terms of voltage trace, and firing statistics such as the gain curve and the distribution of inter-spike intervals. Figure 4.4B shows a ten-fold reduction in error for the *full voltage trace* scheme versus the *linear integration* scheme both in terms of error per time step and the accumulated error carried to the time of first spike. From Figure 4.4C, we can see that both the *full voltage trace* and *partial voltage trace* schemes closely follow the gain curve from the NEURON simulation, while without the second-order correction, the *linear integration* scheme doubles the firing rate. The distribution of inter-spike intervals in Figure 4.4D shows a significantly higher peak for the *linear integration* scheme while the other two bilinear integration schemes are comparable with the NEURON simulation, especially at the tail. We show an excerpt of raster plot with Poisson input rate of 95 Hz in Figure 4.4E, where both of the bilinear integration schemes match most of the spikes with the NEURON simulation, while the *linear integration* scheme fires excessively.

5. Discussion

Through asymptotic analysis, we derive the bilinear dendritic integration rule for a neuron that receives multiple synaptic inputs on its dendrites. We then develop fast algorithms based on the rule to simulate the voltage dynamics of a passive neuron with realistic dendritic morphology. The performance of the two bilinear integration schemes in the algorithms show a massive reduction in terms of computational cost, while retaining a relatively high accuracy, compared with the *linear integration* scheme. Although the algorithms are evaluated only with few synaptic locations with large input strengths here, the algorithms are expected to be applicable to simulate real neurons in the brain. In fact, it is quite common for a cortical neuron to have a log-normal distribution of the amplitudes of its presynaptic EPSPs [3]. Therefore, there are only a handful of bilinear interactions that are large enough to warrant attention [6] since the size of the bilinear correction is proportional to the product of the two single-input voltage responses at soma.

5.1. Sources of numerical error. Higher orders (≥ 3) may come into play when the synaptic input rate is very high. For example, suppose there are n excitatory synaptic inputs with the same amplitude of A that arrive within an interval of ~ 10 ms, with a typical second-order coefficient $k_{EE} \sim 0.1$ mV^{-1} for each pair. This interval of ~ 10 ms is chosen so that for a given time during this interval all the single-input responses are at a relatively high voltage level, $\sim 0.9A$. Then the bilinear contribution (for a total of $n(n-1)$ input pairs) increases with $\mathcal{O}(n^2)$. Thus, the ratio γ of the bilinear contributions to the linear contribution can be approximately obtained as follows,

$$\gamma \sim \frac{-0.1(n(n-1)/2)(0.9A)^2}{0.9An} = \frac{-9A(n-1)}{200}. \tag{5.1}$$

In this case, the total response could actually be reduced to zero or below because we can have stronger inhibitory effect from the bilinear interaction than the total sum of linear responses when $\gamma < -1.0$, e.g., when $n \geq 24$ and $A = 1$ mV. This is unrealistic as such nonlinearity is not observed experimentally in pyramidal neurons. The combined synaptic input rate in this case is 2400 Hz. For a total of six strong excitatory synapses as we implemented here, this would correspond to an instantaneous input rate of ~ 400 Hz for each synapse, which is a quite extreme case. However, if we increase the number of synapses to ~ 100, then it corresponds to a 24 Hz instantaneous input rate, well within the physiological value. Thus, the higher order contributions are quite important when considering more synaptic inputs with strong input strength or higher input rates. In addition to the error from dropping the higher-order terms, the numerical error of the arithmetic schemes, i.e., full and partial voltage trace schemes, largely comes from the linear interpolation of data in the library, instead of the time step since the library is built from the simulation that uses the same time step.

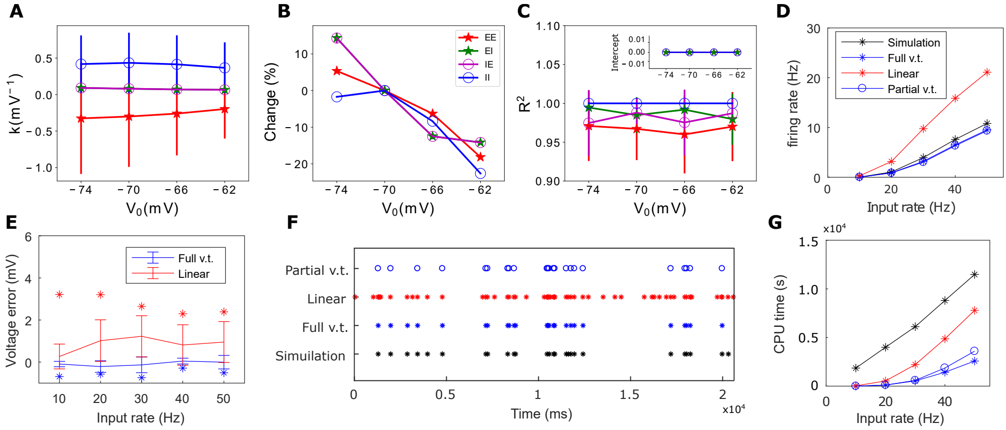

Fig. 5.1. *The performance of the fast algorithms for a semi-active neuron: (A), (B), and (C) show the statistics for the two-input dendritic integration rule on a semi-active neuron. (A) Plots of the second-order coefficient k for all pairs of synaptic inputs as a function of v_0, where the pairs are categorized into combinations of EE, EI, IE, and II. The errorbars give the standard deviations for different combinations of synaptic locations. The coefficient is evaluated for each pair of synapses at the maximum of the product of their responses $|\widehat{V}_p\widehat{V}_q|$ when receiving the two synaptic inputs simultaneously, $\Delta t = 0$. Because $\Delta t = 0$ ms, k_{EI} overlaps with k_{IE}. (B) Plots of the percentage of change in k, with respect to the k at $v_0 = V_r$, across different v_0 for data in (A). The errorbars are too small to be visible. (C) Plots of the goodness of fit R^2 at different v_0. The errorbars show the mean and averaged standard deviation of $\Delta t \in \{0,4,8,12\}$ ms, $t \in \{4,8,12,20,30,50,70,130\}$ ms and all combinations of pairs of synapses. The inset figure shows the intercept of the fitted lines. (D), (E), (F), and (G) plot the statistics of data from four trials of 124 seconds, synaptic inputs are generated from the independent Poisson processes for each synapse. (D) Gain curves for each scheme with the Poisson input rate to the synapses as the abscissa. (E) The mean and standard deviations of the difference between the voltage traces of the simulation (NEURON software) and those of the two schemes (errorbars), i.e., linear integration scheme in red, and full voltage trace (full v.t.) scheme in blue, per time step before the first spike. The accumulated errors at the last time step before the first spike (asterisks). (F) An excerpt of raster plot of the spike timings, with the same Poisson input rate of* 20 Hz *for each scheme. (G) Computational costs, in terms of CPU time spent, against the Poisson input rate for the four different schemes, the legend follows the one in (D).*

5.2. Performance of the algorithms on a semi-active neuron. The fast algorithms can also be applied to neurons with semi-active dendrites and an active soma capable of generating full-shape spikes. We set a soft threshold for the voltage at the soma. When its membrane potential crosses the threshold, instead of resetting the membrane potential across the neuron, we hand over the simulation from the fast algorithms to the NEURON software, which either continues to produce a spike shape and hyperpolarizes or signals a false alarm without spiking and crosses back from the threshold. For both cases, we then hand back the control to the fast algorithms. The dendrites are semi-active in a sense that they preserve many commonly observed active ion channels (including the delayed rectifier potassium channel, two variants of the A-type potassium channel, and the hyperpolarization activated channel), except for the voltage-gated sodium channels on the apical dendrites. The sodium channels result in complex nonlinear effects, to which the dendritic integration rule derived from the passive cable equation (Equation (2.5)) is not directly applicable. The specific nonlinear factors that could substantially affect the numerical accuracy of the algorithms include dendritic spikes and spiking threshold that sensitively depends on the voltage distribution of the neuron.

Figure 5.1A, B and C show that the two-input dendritic integration rule is still valid for the case of the semi-active neuron. Figure 5.1A gives the values of second-order coefficient k across different initial voltage, which are generally larger than the passive neuron in absolute values under the influence of the active channels. This implies that the error from omitting the higher-order terms may have a larger role than that in the passive case. In Figure 5.1B, we can see a smaller change of k in percentage (with respect to k at $v_0 = V_r$) compared with the result of the passive neuron. In addition, different from the passive case, we can observe a significant decrease in the absolute value of second-order coefficient k_{EE} as v_0 increases. This can be explained by the activation of the sodium channel at the soma, which facilitates the depolarization of the membrane potential. Thus, the extra depolarization from the sodium channel activation cancels out some of the hyperpolarizing effect from the bilinear integration, effectively reducing the amplitude of k_{EE}. Figure 5.1C and its inset show the consistently good fit of the second-order coefficient as R^2 is close to 1.0 and the intercept stays zero.

In Figure 5.1D and the bottom panel of Figure 5.1, we show the performance of the *full voltage trace*, *partial voltage trace* and *linear integration* schemes applied on the semi-active neuron, compared with direct numerical computation of the semi-active cable neuron model by the NEURON software. Figure 5.1E shows that the *full voltage trace* scheme still achieves higher accuracy than the *linear integration* scheme, in terms of both the error per time step in the sub-threshold voltage range and accumulated error before spiking. A series of hierarchical linear-nonlinear (hLN) models has been introduced by Ujfalussy et al., 2018 [29]. The performances of the hLN models show that a single-layer (linear integration) hLN model's explanatory power of the sub-threshold voltage does not degenerate much ($\sim 4\%$) when active channels are involved. Consistently, in our model, comparing Figure 5.1E and Figure 4.4B from semi-active and passive neuron, respectively, we find that the sub-threshold voltage error of the linear scheme is about the same in both cases. In our scenario, this indicates that the nonlinearities within the ion channels (as long as there's no dendritic spikes) are not as influential as the nonlinearities resulting from bilinear integrations.

In Figure 5.1F, we can see that the spike timings obtained from both the *full voltage trace* and the *partial voltage trace* schemes match well with those simulated using the NEURON software. From the gain curves plotted in Figure 5.1D, the *linear integration* scheme fires a substantially larger number of spikes than the others, while the firing rates obtained from the two *bilinear integration* schemes only lag a little behind the NEURON simulation result, although they cannot achieve equally good performance as in the case with passive neuron. A small tendency of separation can be observed between the firing rate of result from the NEURON software and that of the bilinear integration schemes as the Poisson input rate increases. Finally, in Figure 5.1G, the computational cost is plotted against the Poisson input rate. Since the spike shapes are all simulated by the NEURON software, it is conceivable to observe a much smaller decrease in terms of CPU time saved compared to the spike-reset dynamics. Indeed, as the input rate increases, following the increasing number of spikes, especially for the *linear integration* scheme (Figure 5.1F and G), the computational cost rises quickly.

In summary, we have developed fast algorithms based on the bilinear dendritic integration rule derived from the passive cable model, with a dependency on the voltage of the soma when inputs arrive. Our algorithms can also achieve reasonably good performance when applied to a neuron with an active soma and semi-active dendrites. For future studies on the subject, it is important to investigate the dependency of the dynamic spiking threshold on the voltage distribution of an active neuron in order to build a spike shape library to further improve the computational efficiency. For a more general application, it is also crucial to explore integration rules for an active neuron with dendritic sodium channels.

Acknowledgment. The authors dedicate this work to Professor David Cai. This work is funded by National Natural Science Foundation of China Grants 11901388, Shanghai Sailing Program 19YF1421400, Shanghai Chengguang program (S.L.), and Shanghai Rising-Star Program 15QA1402600, Natural Science Foundation of China Grants 11671259, 11722107, 91630208 (D.Z.), and by Student Innovation Center at Shanghai Jiao Tong University.

REFERENCES

[1] B.V. Atallah and M. Scanziani, *Instantaneous modulation of gamma oscillation frequency by balancing excitation with inhibition*, Neuron, 62(4):566–577, 2009. 1

[2] T. Branco, B.A. Clark, and M. Häusser, *Dendritic discrimination of temporal input sequences in cortical neurons*, Science, 329(5999):1671–1675, 2010. 1

[3] G. Buzsáki and K. Mizuseki, *The log-dynamic brain: how skewed distributions affect network operations*, Nat. Rev. Neurosci., 15(4):264–278, 2014. 5

[4] R.C. Cannon, D.A. Turner, G.K. Pyapali, and H.V. Wheal, *An on-line archive of reconstructed hippocampal neurons*, J. Neuro. Meth., 84(1):49–54, 1998. 4

[5] M.J. Chacron, *Nonlinear information processing in a model sensory system*, J Neurophysiol., 95(5):2933–2946, 2006. 1

[6] L. Cossell, M.F. Iacaruso, D.R. Muir, R. Houlton, E.N. Sader, H. Ko, S.B. Hofer, and T.D. Mrsic-Flogel, *Functional organization of excitatory synaptic strength in primary visual cortex*, Nature, 518(7539):399–403, 2015. 5

[7] F. David, C. Linster, and T.A. Cleland, *Lateral dendritic shunt inhibition can regularize mitral cell spike patterning*, J. Comput. Neurosci., 25(1):25–38, 2008. 1

[8] P. Dayan and L. Abbott, *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*, Cambridge: MIT Press, 2001. 2

[9] A. Destexhe, M. Rudolph, and D. Paré, *The high-conductance state of neocortical neurons in vivo*, Nat. Rev. Neurosci., 4(9):739–751, 2003. 1, 2

[10] F. Gabbiani, H.G. Krapp, C. Koch, and G. Laurent, *Multiplicative computation in a visual neuron sensitive to looming*, Nature, 420(6913):320–324, 2002. 1

[11] L.N. Groschner, L.C.W. Hak, R. Bogacz, S. DasGupta, and G. Miesenböck, *Dendritic integration of sensory evidence in perceptual decision-making*, Cell, 173(4):894–905, 2018. 1

[12] J. Hao, X. Wang, Y. Dan, M. Poo, and X. Zhang, *An arithmetic rule for spatial summation of excitatory and inhibitory inputs in pyramidal neurons*, Proc. Natl. Acad. Sci. USA, 106(51):21906–21911, 2009. 1, 2.1, 4.1.2

[13] M. Hines, A.P. Davison, and E. Muller, *Neuron and python*, Front. Neuroinform., 3:1, 2009. 4

[14] M.P. Jadi, B.F. Behabadi, A. Poleg-Polsky, J. Schiller, and B.W. Mel, *An augmented two-layer model captures nonlinear analog spatial integration effects in pyramidal neuron dendrites*, Proc. IEEE, 102(5):782–798, 2014. 1

[15] Y. Katz, V. Menon, D.A. Nicholson, Y. Geinisman, W.L. Kath, and N. Spruston, *Synapse distribution suggests a two-stage model of dendritic integration in CA1 pyramidal neurons*, Neuron, 63(2):171–177, 2009. 1

[16] C. Koch, *Biophysics of Computation: Information Processing in Single Neurons*, Oxford University Press, 2004. 2, 2.1

[17] C. Koch, T. Poggio, and V. Torre, *Retinal ganglion cells: a functional interpretation of dendritic morphology*, Phil. Trans. R. Soc. Lond. B, 298(1090):227–263, 1982. 1

[18] S. Li, N. Liu, X-H. Zhang, D. Zhou, and D. Cai, *Bilinearity in spatiotemporal integration of synaptic inputs*, PLoS Comput. Biol., 10(12):e1004014, 2014. 1, 4.1.2

[19] S. Li, D. Zhou, and D. Cai, *Analysis of the dendritic integration of excitatory and inhibitory inputs using cable models*, Commun. Math. Sci., 13(2):565–575, 2015. 1

[20] M. London and M. Häusser, *Dendritic computation*, Annu. Rev. Neurosci., 28:503–532, 2005. 1

[21] J.C. Magee, *Dendritic integration of excitatory synaptic input*, Nat. Rev. Neurosci., 1(3):181–190, 2000. 1

[22] M. Megıas, Z.S. Emri, T.F. Freund, and A.I. Gulyas, *Total number and distribution of inhibitory and excitatory synapses on hippocampal CA1 pyramidal cells*, Neuroscience, 102(3):527–540, 2001. 4.1

[23] P. Poirazi, T. Brannon, and B.W. Mel, *Arithmetic of subthreshold synaptic summation in a model CA1 pyramidal cell*, Neuron, 37(6):977–987, 2003. 1

[24] P. Poirazi, T. Brannon, and B.W. Mel, *Pyramidal neuron as two-layer neural network*, Neuron, 37(6):989–999, 2003. 1

[25] C. Schmidt-Hieber, G. Toleikyte, L. Aitchison, A. Roth, B.A. Clark, T. Branco, and M. Häusser, *Active dendritic integration as a mechanism for robust and precise grid cell firing*, Nat. Neurosci., 20(8):1114-1128, 2017. 1

[26] N. Spruston, G. Stuart, and M. Häusser, *Principles of dendritic integration*, Dendrites, 351:597, 2016. 1

[27] G.J. Stuart and N. Spruston, *Dendritic integration: 60 years of progress*, Nat. Neurosci., 18(12):1713–1721, 2015. 1

[28] H.C. Tuckwell, *Introduction to Theoretical Neurobiology: Volume 1, Linear Cable Theory and Dendritic Structure*, Cambridge University Press, 1, 1988. 2

[29] B.B. Ujfalussy, J.K. Makara, M. Lengyel, and T. Branco, *Global and multiplexed dendritic computations under in vivo-like conditions*, Neuron, 100(3):579–592, 2018. 5.2

[30] C. Van Vreeswijk and H. Sompolinsky, *Chaos in neuronal networks with balanced excitatory and inhibitory activity*, Science, 274(5293):1724–1726, 1996. 1, 2

[31] I. Vida, M. Bartos, and P. Jonas, *Shunting inhibition improves robustness of gamma oscillations in hippocampal interneuron networks by homogenizing firing rates*, Neuron, 49(1):107–117, 2006. 1