J Comput Neurosci
DOI 10.1007/s10827-010-0309-5

Reduced order modeling of passive and quasi-active dendrites for nervous system simulation

Boyuan Yan · Peng Li

Received: 7 June 2010 / Revised: 28 October 2010 / Accepted: 20 December 2010

Abstract Accurate neuron models at the level of the single cell are composed of dendrites described by a large number of compartments. The network-level simulation of complex nervous systems requires highly compact yet accurate single neuron models. We present a systematic, numerically efficient and stable model order reduction approach to reduce the complexity of large dendrites by orders of magnitude. The resulting reduced dendrite models match the impedances of the full model within the frequency range of biological signals and reproduce the original action potential output waveforms.

Keywords Passive dendrites · Quasi-active dendrites · Reduced modeling · Computer simulation

1 Introduction

Brain activities are controlled by large networks of neurons which process information in the form of electrical signals passed from one neuron to another via synaptic connections. When coupled with experimental studies, numerical simulation of large scale nervous systems can provide profound understanding of brain functions, offering insights that are practically or ethically impossible to acquire purely through experimental means (Markram 2006; Izhikevich and Edelman 2007; Djurfeldt et al. 2007).

A fundamental function of neuron cells is to transform the incoming synaptic information into specific patterns of action potential output. This process is known as synaptic integration and is performed in the extensively branched dendritic tree of the neuron cell. Therefore, to understand information processing in complex neuron networks, it is imperative to develop simplified dendrite models, which captures the input-output characteristics of the full models. In the past thirty years, a lot of efforts have been devoted to fulfilling this goal (Wilson and Bower 1989; Segev 1992; Rapp et al. 1992; Bush and Sejnowski 1993). Most of such existing methods apply a set of rules derived from the principles of cable theory (Rall 1959) to collapse the full dendritic tree into a morphologically simpler canonical representation and map the electrical and synaptic properties from the full tree onto the simpler representation. While being successful in preserving some characteristics of the full model, those methods have some significant limitations. The model simplification process adopted in the above approaches is often very ad-hoc and lacks systematic error control.

Recently, a balanced truncation (BT) method (Moore 1981) was proposed as a benchmark method for quasi-active reduction (Kellems et al. 2009). Quasi-active systems are obtained by linearizing a nonlinear neuron at the resting potential. Those models are useful for the study of subthreshold membrane potential and the analysis of noise in dendritic information processing (Koch 1999). The underlying computational machinery of the proposed approach is model order reduction (MOR) (Moore 1981; Glover 1984; Pillage and Rohrer

Action Editor: James M. Bower

B. Yan (✉) · P. Li
Department of Electrical and Computer Engineering,
Texas A&M University, College Station, TX 77843, USA
e-mail: byan@tamu.edu, boyuanyan@gmail.com

P. Li
e-mail: pli@tamu.edu

Published online: 12 January 2011

1990; Feldmann and Freund 1995; Grimme et al. 2005; Odabasioglu 1998; Bai and Su 2005; Antoulas 2005; Yan et al. 2008), which computes a reduced order model for a given higher-order linear time invariant (LTI) system. As BT is very expensive, another method based on H_2 approximation, IRKA (Gugercin et al. 2008), was proposed for large-scale systems to mitigate the computational cost. While these methods have shown good potentials in reducing dendrites and posses favorable theoretical properties, they are often limited by high computational cost and complexity in implementation.

Although the active properties of dendrites have been widely studied, passive dendrite models are still valuable under many cases. Interestingly, from a computational point of view, compared with quasi-active dendrite reduction, passive dendrite reduction is even more challenging. While quasi-active dendrite is represented by a multi-input-single-output (MISO) LTI system, passive dendrite should be represented by a multi-input-multi-output (MIMO) LTI system. The efficiency of MOR techniques for LTI systems degrades dramatically for MIMO systems with a large number of inputs and outputs. In addition, it is important to preserve passivity when reducing passive dendrite as non-passive reduced model can produce non-physical oscillation in simulation when interconnected with other elements (like active conductances) to form a larger network. Therefore, in spite of high computational cost, existing techniques proposed for quasi-active dendrite reduction can not be applied for passive dendrite reduction.

In this paper, we propose computationally efficient Krylov subspace based MOR techniques (Feldmann and Freund 1995; Grimme et al. 2005; Odabasioglu 1998; Bai and Su 2005; Yan et al. 2008). Those methods are based on the concept of moment-matching and have gained success for reduced order modeling of large-scale RLC networks in the field of electrical engineering. Given a multi-compartment neuron model, we first isolate active ion channels from the neuron cell and then model the remaining passive dendrite as an N-port linear circuit consisting of a large number of RC compartments. Such N-port dendrite circuit is internally represented by a high-order impedance matrix, which is considered as a canonical full model. The full model is then reduced by a numerically efficient and stable Krylov subspace based MOR technique, which produces a much smaller impedance representation, or, the reduced order model.

The reduced order model not only matches the frequency-domain characteristics of the full model but also preserves the passivity of the original model (Odabasioglu 1998). As the spectrum of neuron activities is band limited and has a significant component towards the low frequency end, the proposed method is very efficient in terms of both computational cost and the size of reduced model as only a small number of moments are needed. In the case of integration of a large number of synaptic inputs, we further develop a scheme to exploit the inherent signal attenuation in the dendrite tree to effectively reduce the number of ports that need to be considered. This noticeably reduces the complexity of MOR, and increases the compactness of the reduced order model. In addition to passive dendrites, the proposed method has also been extended for reduction of quasi-active systems. For quasi-active dendrite models, we recognize that such systems can be effectively modeled as an MISO system and exploit such property for efficient model order reduction. In particular, we develop an output Krylov subspace method that can compute highly compact reduced models very efficiently. Our experimental results have demonstrated that the proposed methods are more accurate, less expensive and easier to implement than the existing methods.

2 Method

In Section 2.1, we use a simple example to illustrate how a passive dendrite tree can be modeled at the input and output ports using an N-port impedance. Then we present Krylov subspace based model reduction techniques to reduce the above N-port model in Section 2.2. In Section 2.3, the proposed methods are extended for the reduction of quasi-active dendrite. Section 2.4 gives the details of model parameters and simulations protocols.

2.1 Equivalent N-port circuits for passive dendrites

As illustrated in Fig. 1, a realistic dendrite tree can be modeled as a set of cylindrical membrane cables (Rall 1959) and each cylinder can be divided into a number of identical compartments of length l based on the compartment approach (Rall 1964). If the length l is sufficiently small, each compartment can be assumed to be isopotential and the compartment approach results in equivalent circuit models for the neurons, which makes it possible to apply the well developed circuit simulation techniques for neuron simulation.

In this section, we use a simple example to illustrate how a given full dendrite tree can be modeled at the input and output ports using an N-port impedance. As shown in Fig. 2, without loss of generality, a neuron

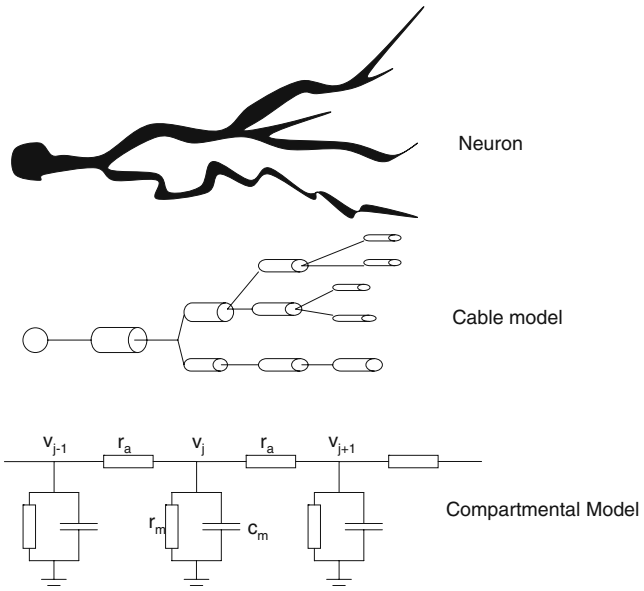

Fig. 1 Cable and compartment models for dendrites

consisting of four compartments is used in this illustrative example. The equivalent circuit is partitioned into four portions, which are interconnected by three ports. In circuit theory, a *port* is the location where different portions of circuits are connected and each port is described by the port voltage v_p and current i_p, as shown in the figure.

2.1.1 Somatic voltage gated ion channel

In the illustrative example, the first compartment represents the soma, which is the place where action potentials are initiated. To model action potentials, voltage gated ion channels are connected to the soma at the first port (port 1). In this example, there are three types of active ion channels described by the Hodgkin–Huxley model (Hodgkin and Huxley 1952) and the voltage-current relationship for the ion channels is given as follows

$$i_{p1} = g_{Na}m^3h(E_{Na} - v_{p1}) + g_K n^4(E_K - v_{p1}) + g_L(E_L - v_{p1}) \quad (1)$$

where v_{p1} is the port voltage, the port current i_{p1} is the sum of three ionic currents, E_{Na}, E_K, E_L are reversal potentials, g_{Na}, g_K, g_L are maximum conductance, and m, n, h are gating variables describing the probability that a channel is open and evolving as follows

$$\begin{aligned} \dot{m} &= \alpha_m(v_{p1})(1-m) - \beta_m(v_{p1})m \\ \dot{n} &= \alpha_n(v_{p1})(1-n) - \beta_n(v_{p1})n \\ \dot{h} &= \alpha_h(v_{p1})(1-h) - \beta_h(v_{p1})h \end{aligned} \quad (2)$$

The above equations indicate that the voltage-current relationship of voltage gated ion channels is nonlinear.

2.1.2 Synaptically activated ion channel

There is a synaptic input connected to the second compartment at the second port (port 2). The synaptic input is modeled as a synaptically activated ion channel and the voltage-current relationship is given as follows

$$i_{p2} = g_{syn}(t)(E_{syn} - v_{p2}), \quad (3)$$

where v_{p2} is the port voltage, i_{p2} is the port current, E_{syn} is the reversal potential, and $g_{syn}(t)$ is the time-dependent conductance, which is often approximated by an analytic alpha function (Rall 1967; Jack et al. 1975)

$$g_{syn}(t) = g_{\max}\frac{t - t_0}{t_p}e^{(1-(t-t_0)/t_p)}(t > t_0) \quad (4)$$

Given the arrival time t_0 of a spike from a presynaptic neuron, g_{syn} increases rapidly to a maximum conductance $g_{\max}$ at $t = t_0 + t_p$ and decreases more slowly to zero afterwards.

2.1.3 Electrode

There is an electrode connected to the last compartment at the third port (port 3). The electrode is described by an independent current source, which means the port current i_{p3} is only a function of time t and independent of the port voltage v_{p3}

$$i_{p3} = f(t) \quad (5)$$

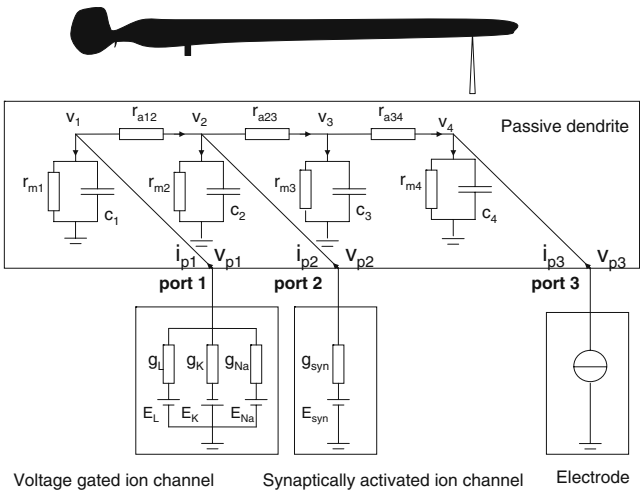

Fig. 2 The equivalent circuit of a neuron consisting of four compartments

2.1.4 An equivalent multiport linear circuit model for the passive dendrite

The previous three portions (voltage gated ion channels, synaptically activated ion channel, and electrode) are all active. Isolating those active portions from the neuron, the rest part of the neuron is the passive dendrite, which is modeled as an equivalent multiport linear circuit. The dendrite sub-circuit interfaces other active portions of the neuron at the three ports mentioned before. To characterize the dendrite as a sub-circuit of the overall neuron circuit, we derive the voltage-current relations of this 3-port linear circuit at the ports by relating the port currents (i_{p1}, i_{p2}, i_{p3}) with the port voltages (v_{p1}, v_{p2}, v_{p3}).

As shown in Fig. 2, c_1, c_2, c_3, c_4 are membrane capacitors, $r_{m1}, r_{m2}, r_{m3}, r_{m4}$ are membrane resistors, $r_{a12}, r_{a23}, r_{a34}$ are axial resistors, and v_1, v_2, v_3, v_4 are membrane potentials. Note that, in the circuit models in this paper, all voltages are expressed with respect to a resting potential that is defined to be $0V$. Given the port currents i_{p1}, i_{p2}, i_{p3}, it is easy to verify the compartments 1–4 can be described by the following equations correspondingly ($c_1 = c_2 = c_3 = c_4 = c$, $r_{m1} = r_{m2} = r_{m3} = r_{m4} = r_m$, $r_{a12} = r_{a23} = r_{a34} = r_a$ for simplicity)

$$\begin{aligned}
c\dot{v}_1 + \frac{v_1}{r_m} + \frac{v_1 - v_2}{r_a} &= i_{p1} \\
c\dot{v}_2 + \frac{v_2}{r_m} + \frac{v_2 - v_3}{r_a} &= \frac{v_1 - v_2}{r_a} + i_{p2} \\
c\dot{v}_3 + \frac{v_3}{r_m} + \frac{v_3 - v_4}{r_a} &= \frac{v_2 - v_3}{r_a} \\
c\dot{v}_4 + \frac{v_1}{r_m} &= \frac{v_3 - v_4}{r_a} + i_{p3}
\end{aligned} \tag{6}$$

which can be written in a matrix form

$$\begin{bmatrix} c & 0 & 0 & 0 \\ 0 & c & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & c \end{bmatrix} \begin{bmatrix} \dot{v}_1 \\ \dot{v}_2 \\ \dot{v}_3 \\ \dot{v}_4 \end{bmatrix}$$

$$= -\begin{bmatrix} 1/r_m + 1/r_a & -1/r_a & 0 & 0 \\ -1/r_a & 1/r_m + 2/r_a & -1/r_a & 0 \\ 0 & -1/r_a & 1/r_m + 2/r_a & -1/r_a \\ 0 & 0 & -1/r_a & 1/r_m + 1/r_a \end{bmatrix}$$

$$\times \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} i_{p1} \\ i_{p2} \\ i_{p3} \end{bmatrix} \tag{7}$$

The port voltages v_{p1}, v_{p2}, and v_{p3} are the compartment voltages v_1, v_2, and v_4, respectively, which can be obtained from the following equation

$$\begin{bmatrix} v_{p1} \\ v_{p2} \\ v_{p3} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} \tag{8}$$

As a result, the voltage-current relations of the 3-port linear circuit in the illustrative example are described by Eqs. (7) and (8). Generally, the voltage-current relations of a passive dendrite with n compartments and $N (N \leq n)$ ports can be described by the following matrix equations

$$\begin{aligned}
\mathbf{C}\dot{\mathbf{v}} &= -\mathbf{G}\mathbf{v} + \mathbf{B}\mathbf{i}_p \\
\mathbf{v}_p &= \mathbf{B}^T\mathbf{v}
\end{aligned} \tag{9}$$

which is the state-space model for the equivalent N-port linear circuit. The vectors $\mathbf{v} \in \mathbb{R}^{n\times 1}$, $\mathbf{i}_p \in \mathbb{R}^{N\times 1}$, and $\mathbf{v}_p \in \mathbb{R}^{N\times 1}$ denote the compartment voltages, port currents, and port voltages, respectively

$$\mathbf{v} = \begin{bmatrix} v_1 \\ \cdots \\ v_j \\ \cdots \\ v_n \end{bmatrix} \mathbf{i}_p = \begin{bmatrix} i_{p1} \\ \cdots \\ i_{pk} \\ \cdots \\ i_{pN} \end{bmatrix} \mathbf{v}_p = \begin{bmatrix} v_{p1} \\ \cdots \\ v_{pk} \\ \cdots \\ v_{pN} \end{bmatrix} \tag{10}$$

The matrices $\mathbf{C} \in \mathbb{R}^{n\times n}$ and $\mathbf{G} \in \mathbb{R}^{n\times n}$ represent the capacitance and conductance matrices. Note that, both $\mathbf{C}$ and $\mathbf{G}$ are symmetric. The matrix $\mathbf{B} \in \mathbb{R}^{n\times N}$ is the incidence matrix and $\mathbf{B}^T$ is the transpose of $\mathbf{B}$. The incidence matrix $\mathbf{B}$ maps the ports to the compartments in the following way

$$\begin{aligned}
&\mathbf{B}_{jk} = 1 \text{ the } k\text{th port is in the } j\text{th compartment} \\
&\mathbf{B}_{jk} = 0 \text{ otherwise}
\end{aligned} \tag{11}$$

where $j = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, N$.

2.2 Reduced order models of passive dendrites

The number of differential equations n is the *order* of the state-space model (9). In Fig. 2, the order is 4. However, for a realistic dendrite tree with thousands of compartments, the order is high and so is the computational cost involved in analyzing such large dendrites in the time-domain simulation.

As illustrated in Fig. 3, an efficient simulation scheme to mitigate this problem is to preprocess the large linear portion of the circuit (passive dendrite) into a reduced order multiport model with similar voltage-current relations (port behavior). Then the reduced

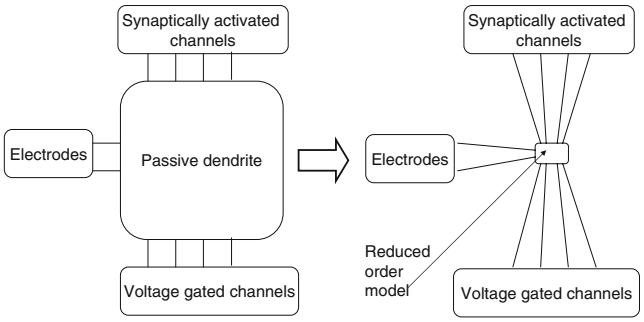

Fig. 3 Preprocessing of linear portions of circuits (passive dendrites) into N-port reduced models

model is used to simulate with other portions of the circuit (voltage gated ion channels, synaptically activated ion channels, and electrodes), which are active and could be nonlinear.

In this section, we introduce the concept of model reduction and then apply Krylov subspace methods to generate N-port reduced order models for passive dendrites.

2.2.1 Model order reduction

By taking the Laplace transform of the state-space model (9), the voltage-current relations of the N-port linear circuit can be obtained in the complex frequency domain

$$\mathbf{v}_p(s) = \mathbf{Z}(s)\mathbf{i}_p(s) \tag{12}$$

where $\mathbf{Z}(s) \in \mathbb{C}^{N\times N}$ is the Z-parameter matrix with impedance parameters for the N-port circuit

$$\mathbf{Z}(s) = \mathbf{B}^T(s\mathbf{C} + \mathbf{G})^{-1}\mathbf{B}, \tag{13}$$

$$\begin{bmatrix} v_{p1}(s) \\ \vdots \\ v_{pN}(s) \end{bmatrix} = \begin{bmatrix} z_{11}(s) & \cdots & z_{1N}(s) \\ \vdots & & \vdots \\ z_{N1}(s) & \cdots & z_{NN}(s) \end{bmatrix} \begin{bmatrix} i_{p1}(s) \\ \vdots \\ i_{pN}(s) \end{bmatrix}$$

$\mathbf{v}_p(s)$ $\quad$ $\mathbf{Z}(s)$ $\quad$ $\mathbf{i}_p(s)$

Each entry of $\mathbf{Z}(s)$ is a rational function of order n

$$z_{ij}(s) = \frac{b_1 s^{n-1} + b_2 s^{n-2} + \cdots + b_n}{s^n + a_1 s^{n-1} + a_2 s^{n-2} \cdots + a_n}$$

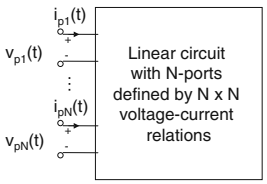

Fig. 4 The multiport representation of a linear circuit in the complex frequency domain

which is the function of complex frequency variable $s \in \mathbf{C}$.

As shown in Fig. 4, $\mathbf{Z}(s)$ is a $N \times N$ matrix and each element $z_{ij}(s)(i, j = 1, 2, \ldots, N)$ is a rational function of order n in the complex frequency domain. If $i = j$, $z_{ii}(s)$ is the input impedance of port i; if $i \neq j$, $z_{ij}(s)$ is the transfer impedance from port j to port i.

Given the state-space model (9), we want to produce a much smaller q-th order model

$$\begin{aligned} \tilde{\mathbf{C}}\dot{\tilde{\mathbf{v}}} &= -\tilde{\mathbf{G}}\tilde{\mathbf{v}} + \tilde{\mathbf{B}}\mathbf{i}_p \\ \tilde{\mathbf{v}}_p &= \tilde{\mathbf{B}}^T\tilde{\mathbf{v}} \end{aligned} \tag{14}$$

where $\tilde{\mathbf{C}}, \tilde{\mathbf{G}} \in \mathbb{R}^{q\times q}$, and $\tilde{\mathbf{B}} \in \mathbb{R}^{q\times N}$.

Order q is much smaller than the original order n, i.e. $q \ll n$, but the output $\mathbf{v}_p$ and $\tilde{\mathbf{v}}_p$ is approximately equal for inputs $\mathbf{i}_p$ of interest. In order to achieve this, the Z-parameter matrix of reduced model

$$\tilde{\mathbf{Z}}(s) = \tilde{\mathbf{B}}^T(s\tilde{\mathbf{C}} + \tilde{\mathbf{G}})^{-1}\tilde{\mathbf{B}} \tag{15}$$

has to be close to the Z-parameter matrix of original model $\mathbf{Z}(s)$ (13) such that

$$\|\mathbf{Z}(s) - \tilde{\mathbf{Z}}(s)\| < \epsilon \tag{16}$$

in some appropriate norm, for some given allowable error and allowed domain of the complex frequency variable s. Also, we want to make sure the reduced model $\tilde{\mathbf{Z}}(s)$ is also passive.

2.2.2 Krylov subspace reduction method

Model reduction is an efficient technique to reduce the internal complexity of a system while preserving the port behaviors. Recently developed Krylov subspace reduction methods have gained great success in the application to large systems (Feldmann and Freund 1995; Grimme et al. 2005; Odabasioglu 1998; Bai and Su 2005). Those methods are based on the moment matching of Z-parameter matrix (13), which is a matrix-valued rational function and can be expanded as Taylor series at the origin ($s = 0$)

$$\mathbf{Z}(s) = \mathbf{M}_0 + \mathbf{M}_1 s + \cdots + \mathbf{M}_k s^k + \ldots \tag{17}$$

where $\mathbf{M}_k \in \mathbb{R}^{N\times N}$ is the kth moment of Taylor expansion.

We want the reduced Z-parameter matrix $\tilde{\mathbf{Z}}(s)$ to agree with the original Z-parameter matrix $\mathbf{Z}(s)$ up to the first m derivatives (moments) at the expansion point, the origin ($s = 0$)

$$\tilde{\mathbf{M}}_k = \mathbf{M}_k(k = 0, 1\ldots., m-1) \tag{18}$$

To illustrate the effects of moment matching, we use $z_{ij}(s)$, one entry of the matrix $\mathbf{Z}(s)$, as an example to show its frequency response. The frequency response is obtained by evaluating the magnitude of $z_{ij}(s)$ at $s = j\omega$. As shown in Fig. 5, the impedance of the reduced and original models agree around expansion point $\omega = 0$ and the impedance of reduced model will approach the impedance of original model in a wider frequency range as the reduced order increases.

Conceptually, it is not difficult to understand that matching a few original moments in the reduced model will well approximate the frequency domain characteristics of the full impedance model around the expansion point. In practice, it is often also true that moment matching will preserve the original frequency domain responses at frequencies that are far away from the expansion point, as shown in Fig. 5. This salient feature has made moment matching a popular choice for reducing LTI systems. It is possible to directly compute the moments of the full model and match them in the reduced model (Pillage and Rohrer 1990). However, such "direct" approach has been shown to be numerically instable (Feldmann and Freund 1995). Much more robust Krylov subspace projection based methods address this difficulty by computing a subspace spanned by the moment vectors (instead of moment vectors themselves) and projecting the full model onto this subspace (Feldmann and Freund 1995; Grimme et al. 2005; Odabasioglu 1998; Bai and Su 2005). Hence, these methods are numerically stable moment-matching based reduction techniques and the details are as follows.

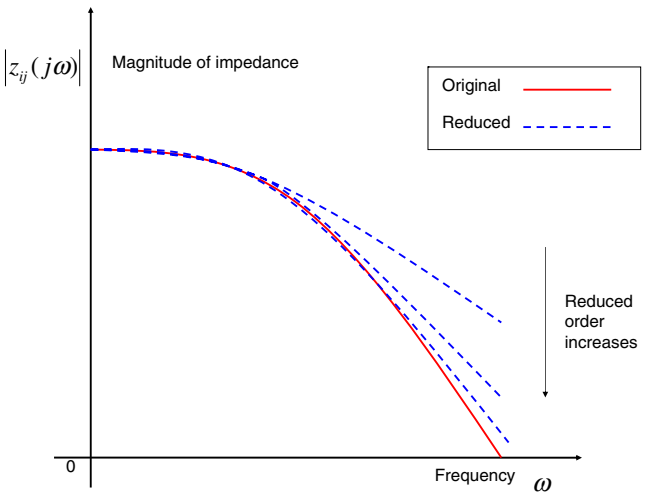

Fig. 5 The moment matching of impedance parameter

The Krylov subspace $K_m(\mathbf{A}, \mathbf{R})$ generated by a matrix $\mathbf{A}$ and a matrix $\mathbf{R}$, of order m, is the subspace spanned by the set of vectors

$$K_m(\mathbf{A}, \mathbf{R}) = span(\mathbf{R}, \mathbf{AR}, \mathbf{A^2R}, \ldots, \mathbf{A^{m-1}R}). \tag{19}$$

Given the state-space model (9), if we construct a projection matrix $\mathbf{V} \in \mathbb{R}^{n \times mN}$, whose columns span a Krylov subspace $K_m(\mathbf{A}, \mathbf{R})$

$$span(\mathbf{V}) = K_m(\mathbf{A}, \mathbf{R}), \tag{20}$$

where

$$\mathbf{A} = \mathbf{G}^{-1}\mathbf{C} \in \mathbb{R}^{n \times n} \mathbf{R} = \mathbf{G}^{-1}\mathbf{B} \in \mathbb{R}^{n \times N}, \tag{21}$$

a reduced model of order $q (q = mN)$ can be obtained by projecting the original equations (9) in the column space of $\mathbf{V}$

$$\tilde{\mathbf{C}} = \mathbf{V}^T\mathbf{C}\mathbf{V}\tilde{\mathbf{G}} = \mathbf{V}^T\mathbf{G}\mathbf{V}\tilde{\mathbf{B}} = \mathbf{V}^T\mathbf{B}. \tag{22}$$

The production process is illustrated in Fig. 6. It has been proved that the reduced model is passive and matches the first m moments of the original model (Odabasioglu 1998).

The projection matrix $\mathbf{V}$ can be obtained by a numerically efficient Arnoldi method. The reduction process is summarized in Algorithm 1. The steps 1–5 are similar to the implementation in Odabasioglu (1998). The last step of Algorithm 1 is to reduce the number of non-zero elements of matrices $\tilde{\mathbf{C}}$ and $\tilde{\mathbf{G}}$. After projection, both matrices are full(all elements are non-zero). To further speed up simulation, we want to reduce the number of nonzero elements. As both matrices are symmetric in the state-space model (9), $\tilde{\mathbf{G}}$ and $\tilde{\mathbf{C}}$ can be simultaneously diagonalized by the following transformation

$$\tilde{\mathbf{C}}_d = \mathbf{W}^T\tilde{\mathbf{C}}\mathbf{W}\tilde{\mathbf{G}}_d = \mathbf{W}^T\tilde{\mathbf{G}}\mathbf{W}\tilde{\mathbf{B}}_d = \mathbf{W}^T\mathbf{B} \tag{23}$$

where $\tilde{\mathbf{C}}_d, \tilde{\mathbf{G}}_d$ are diagonal matrices (off-diagonal entries are all zero), and $\mathbf{W} \in \mathbb{R}^{q \times q}$ is an orthonormal matrix whose columns are the generalized eigenvectors of the matrix pair $(-\tilde{\mathbf{G}}, \tilde{\mathbf{C}})$ (Stewart 2001)

$$\mathbf{W} = eigmatrix(-\tilde{\mathbf{G}}, \tilde{\mathbf{C}}) \tag{24}$$

Note that, the Z-parameter matrix $\tilde{\mathbf{Z}}(s)$ is invariant under the transformation (23).

As the matrices $\mathbf{G}, \mathbf{C}, \mathbf{B}$ are very sparse, the Krylov subspace method is very efficient (Saad 2003). Given

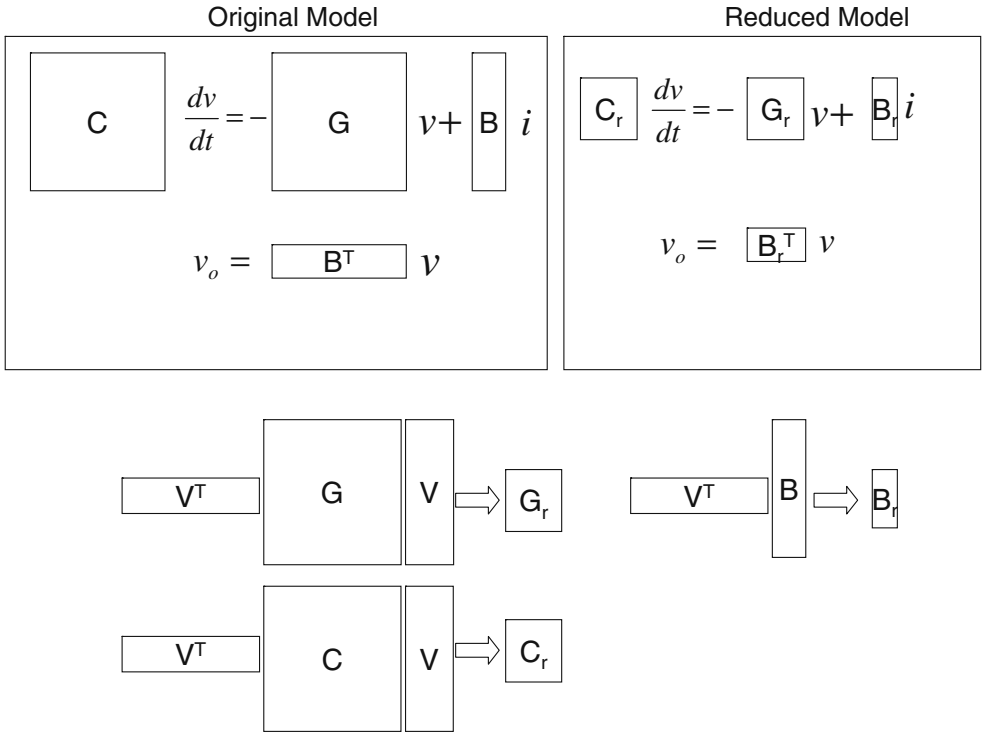

Fig. 6 Model order reduction via projection

the number of compartments n, the computational cost of the reduction algorithm in Algorithm 1 is only dominated by $O(n^r)(1 < r < 2)$, where r is dependent on the sparsity of the matrices. The reduced models have the same impedance parameters as the the full models and can be automatically generated in a inexpensive way. As a result, they could be used to replace the full models to speed up network simulations.

2.2.3 Reduction for dendrites with a large number of ports

Note that, given a system with N ports, to match the first m moments, the dimension of the Krylov subspace will be mN. In this case, the projection matrix $\mathbf{V}$ will have mN columns and the order of reduced system will be $q = mN$.

As shown in Algorithm 1, each port is considered equally in the classical Krylov subspace reduction method. However, if the number of ports is large, the size of reduced model has to be large, which limits the efficiency of model reduction method. In this section, we present an additional scheme to handle dendrites with a large number of ports. Here ports are considered differently in terms of their properties and relative importance:

(1) Among all the ports, the somatic port is the most important one. The voltage responses at

Algorithm 1 Passive order reduction of dendrites

Input: $\mathbf{Z}(s) : (\mathbf{G}, \mathbf{C}, \mathbf{B})\ m$
Output: $\tilde{\mathbf{Z}}(s) : (\tilde{\mathbf{C}}_d, \tilde{\mathbf{G}}_d, \tilde{\mathbf{B}}_d)$

1. Solve $\mathbf{GR} = \mathbf{B}$ for $\mathbf{R}$
2. QR factorization of $\mathbf{R}$: $(\mathbf{X}_0, \mathbf{T}) = qr(\mathbf{R})$
3. For $k = 1, \ldots, m$
   Set $\mathbf{V} = \mathbf{CX}_{k-1}$
   Solve $\mathbf{GX}_k^{(0)} = \mathbf{V}$ for $\mathbf{X}_k^{(0)}$
   For $j = 1, \ldots, k$
   $H = \mathbf{X}_{k-j}^T \mathbf{X}_k^{(j-1)}$
   $\mathbf{X}_k^{(j)} = \mathbf{X}_k^{(j-1)} - \mathbf{X}_{k-j} H$
   QR factorization of $\mathbf{X}_k^{(k)}$: $(\mathbf{X}_k, \mathbf{T}) = qr(\mathbf{X}_k^{(k)})$
4. Set $\mathbf{V} = [\mathbf{X}_0, \mathbf{X}_1, \ldots, \mathbf{X}_{k-1}]$
5. Compute $\tilde{\mathbf{C}} = \mathbf{V}^T\mathbf{CV}$, $\tilde{\mathbf{G}} = \mathbf{V}^T\mathbf{GV}$, $\tilde{\mathbf{B}} = \mathbf{V}^T\mathbf{B}$
6. Diagonalize $\tilde{\mathbf{C}}$ and $\tilde{\mathbf{G}}$ by the transformations $\tilde{\mathbf{C}}_d = \mathbf{W}^T\mathbf{CW}$, $\tilde{\mathbf{G}}_d = \mathbf{W}^T\mathbf{GW}$, $\tilde{\mathbf{B}}_d = \mathbf{W}^T\mathbf{B}$, where $\mathbf{W}$ is the generalized eigenmatrix of the matrix pair $(-\tilde{\mathbf{G}},\tilde{\mathbf{C}})$.

soma represent the final results of information processing via synaptic integration, which are to be passed onto other neurons in the network. In addition, due to the spiking action potentials, somatic signals have more high frequency components than dendritic signals. Therefore, more moments should be matched at the somatic port. In practice, it takes three to five moments for a good approximation.

(2) On the contrary, the performances at other ports are less important and the signals there also have less high frequency components than the signals in the soma. Usually, a good result can be obtained by matching only one moment for each dendritic port.

(3) In addition, as the cable theory predicts, the contributions of inputs to the somatic membrane potentials depend on their locations. In terms of the somatic response, proximal inputs have relatively larger effects than distal inputs, which are attenuated more by the time they reach the action potential initiation zone. For example, given a model of CA1 pyramidal neurons with passive dendrites, an excitatory synapse of fixed synaptic conductance is simulated at different dendrite locations and the amplitude of the corresponding somatic postsynaptic potential is shown in Box 2 in Spruston (2008). As the distal inputs play insignificant roles in the integration process, a good approximation of somatic membrane potential can be obtained even if the moments for distal locations are not matched.

To evaluate the relative importance of each port in terms of the somatic membrane potential quantitatively, we present a metric in terms of electrical distance. If the Z-parameter matrix $\mathbf{Z}(s)$ in Eq. (13) is evaluated at $s = 0$, we get the resistance matrix $\mathbf{R}$

$$\mathbf{R} = \mathbf{Z}(0) = \mathbf{B}^T\mathbf{G}^{-1}\mathbf{B} = \begin{bmatrix} r_{11} & \dots & r_{1p} \\ \dots & \dots & \dots \\ r_{n1} & \dots & r_{np} \end{bmatrix} \tag{25}$$

If a constant current of magnitude I is applied at the jth port ($I_j = I$), a constant voltage response of magnitude $r_{ij}I$ will be generated at the ith port ($V_i = r_{ij}I$). For $i = j$, r_{ij} is the input resistance of the ith port. For $i \neq j$, r_{ij} is the transfer resistance from the jth port to the ith port. For example, as shown in Fig. 7, there are three ports and the first port is at the soma. Given constant DC current inputs of the same amplitude I ($I_1 = I_2 = I_3 = I$), the somatic membrane potential V_1 will be $r_{11}I + r_{12}I + r_{23}I$, which means the entries in the first row of matrix $\mathbf{R}$ indicate the relative importance of inputs at different locations in terms of somatic membrane potential V_1.

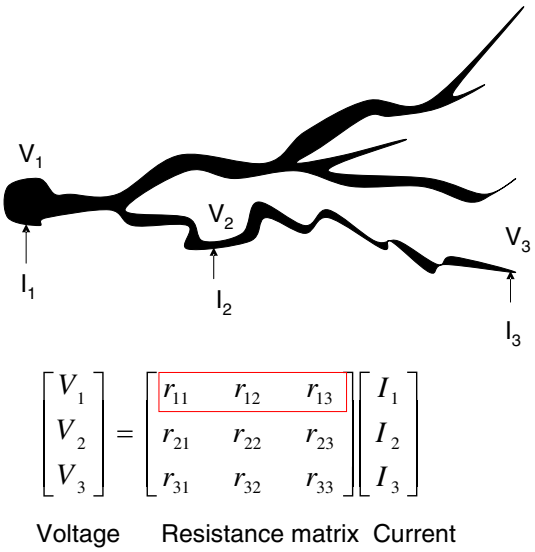

Fig. 7 A metric of synaptic inputs attenuation

Generally, if the kth port is the somatic port, the relative influence of each port on the somatic membrane potential is described by the kth row of the matrix $\mathbf{R}$, $\mathbf{r}_k$, which can be computed as follows

$$\mathbf{r}_k = \mathbf{b}_k^T\mathbf{G}^{-1}\mathbf{B} = \begin{bmatrix} r_{k1} & r_{k2} & \dots & r_{kN} \end{bmatrix} \tag{26}$$

where $\mathbf{b}_k$ is the kth column of input matrix $\mathbf{B}$ corresponding to the kth port. Note that, the equivalent circuits for passive dendrites are RC networks, which behave like low-pass filters. As a result, high frequencies components are attenuated much faster than low frequency components, implying that the resistance parameter is a conservative metric to determine the relative importance of each port ($j = 1, \dots, N$) in terms of the somatic membrane potential. Similar ideas have been successfully applied to reduce large-scale RC networks with massive ports in the field of electrical engineering (Yan et al. 2008).

Based on (1), (2), and (3), we propose a modified Krylov subspace method for dendrites with a large number of ports in Algorithm 2. In the modified algo-

rithm, m moments are matched for the somatic port, one moment is matched for each proximal port, and no moments are matched for distal ports. The proximal ports are identified by evaluating the resistance parameters $\mathbf{r}_k$ given in Eq. (26). Note that, if the percentage of non-somatic ports included as proximal ports is $p(0 \le p \le 1)$, the order of the reduced model will be $m + \lfloor p(N-1) \rfloor$. As shown in Algorithm 2, Steps 1–4 generate a projection matrix $\mathbf{V}$ to match a number of m moments for the somatic port. Steps 5 and 6 identify a percentage of p non-somatic ports as proximal ports based on the evaluation of Eq. (26). In Step 7, one moment is computed for each proximal port and those moments are included into the projection matrix $\mathbf{V}$. The combined projection matrix $\mathbf{V}$ is orthogonalized in Step 8 before projection.

Algorithm 2 Passive order reduction of dendrites with a large number of ports

Input: $\mathbf{Z}(s) : (\mathbf{G}, \mathbf{C}, \mathbf{B})\ m,\ p$
Output: $\tilde{\mathbf{Z}}(s) : (\tilde{\mathbf{C}}_d, \tilde{\mathbf{G}}_d, \tilde{\mathbf{B}}_d)$

1. Solve $\mathbf{Gr} = \mathbf{b}_k$ for $\mathbf{r}$
2. Normalize $\mathbf{v}_0 = \mathbf{r}/||\mathbf{r}||$
3. For $k = 1, \ldots, m$
   Set $\mathbf{v} = \mathbf{C}\mathbf{x}_{k-1}$
   Solve $\mathbf{G}\mathbf{x}_k^{(0)} = \mathbf{v}$ for $\mathbf{x}_k^{(0)}$
   For $j = 1, \ldots, k$
   $H = \mathbf{x}_{k-j}^T \mathbf{x}_k^{(j-1)}$
   $\mathbf{x}_k^{(j)} = \mathbf{x}_k^{(j-1)} - \mathbf{x}_{k-j} H$
   Normalize $\mathbf{v}_k = \mathbf{r}/\left|\left|\mathbf{v}^{(k)}\right|\right|$
4. Set $\mathbf{V} = [\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{k-1}]$
5. Compute $\mathbf{r}_k = \mathbf{b}_k^T \mathbf{G}^{-1} \mathbf{B}$
6. Identify a percentage of p non-somatic ports as proximal ports based on $\mathbf{r}_k$ and generate the proximal input matrix $\mathbf{B}_{proxi}$ which includes the columns of input matrix B corresponding to the proximal ports.
7. Compute the first moment of those inputs $\mathbf{G}^{-1}\mathbf{B}_{proxi}$ and add it to the subspace $\mathbf{V}$ such that $\mathbf{V} = [\mathbf{V}, \mathbf{G}^{-1}\mathbf{B}_{proxi}]$
8. Orthogonalize $\mathbf{V}$ by QR factorization: $(\mathbf{V}, \mathbf{T}) = qr(\mathbf{V})$
9. Compute $\tilde{\mathbf{C}} = \mathbf{V}^T\mathbf{C}\mathbf{V}$, $\tilde{\mathbf{G}} = \mathbf{V}^T\mathbf{G}\mathbf{V}$, $\tilde{\mathbf{B}} = \mathbf{V}^T\mathbf{B}$
10. Diagonalize $\tilde{\mathbf{C}}$ and $\tilde{\mathbf{G}}$ by the transformations $\tilde{\mathbf{C}}_d = \mathbf{W}^T\mathbf{C}\mathbf{W}$, $\tilde{\mathbf{G}}_d = \mathbf{W}^T\mathbf{G}\mathbf{W}$, $\tilde{B}_d = \mathbf{W}^T\mathbf{B}$, where $\mathbf{W}$ is the generalized eigenmatrix of the matrix pair $(-\tilde{\mathbf{G}}, \tilde{\mathbf{C}})$.

It should be noted that although we have suggested a practical way for choosing the number of moments matched for each type of inputs, these values can be adjusted properly to provide tradeoffs between model accuracy, model size, and computational efficiency.

2.3 Model reduction for quasi-active systems

In this section, we first use a simple example to illustrate how a quasi-active dendrite can be represented by a multi-input single-output LTI system. Then we develop efficient Krylov subspace methods for quasi-active dendrite reduction.

2.3.1 Quasi-active compartment models

In the subthreshold regime, the Hodgkin–Huxley system can be linearized about the rest state, which produces a quasi-active model (Koch 1999; Kellems et al. 2009).

In this section, we use an illustrative example to develop a quasi-active neuron model based on a compartment model. As shown in Fig. 8, a neuron consisting of three compartments is used as an illustrative example. In this example, the first compartment represents soma, and the second and third compartments represent the active dendrite.

For simplicity, we assume each compartment only has Na and K active channels, which are described by the Hodgkin–Huxley model. Other active channels can be included in a similar way. It is easy to verify that the compartments can be described by the following equations correspondingly

$$
\begin{aligned}
&c_1\dot{v}_1 + (v_1 - v_2)/r_{12} + i_{Na1} + i_{K1} + i_{L1} = i_{syn1}\\
&c_2\dot{v}_2 + (v_2 - v_3)/r_{23} - (v_1 - v_2)/r_{12} + i_{Na2}\\
&\quad + i_{K2} + i_{L2} = i_{syn2}\\
&c_3\dot{v}_3 - (v_2 - v_3)/r_{23} + i_{Na3} + i_{K3} + i_{L3} = i_{syn3}\\
&\tau_m(v_1)\dot{m}_1 = m_\infty(v_1) - m_1\\
&\tau_m(v_2)\dot{m}_2 = m_\infty(v_2) - m_2\\
&\tau_m(v_3)\dot{m}_3 = m_\infty(v_3) - m_3\\
&\tau_h(v_1)\dot{h}_1 = h_\infty(v_1) - h_1\\
&\tau_h(v_2)\dot{h}_2 = h_\infty(v_2) - h_2\\
&\tau_h(v_3)\dot{h}_3 = h_\infty(v_3) - h_3\\
&\tau_n(v_1)\dot{n}_1 = n_\infty(v_1) - n_1\\
&\tau_n(v_2)\dot{n}_2 = n_\infty(v_2) - n_2\\
&\tau_n(v_3)\dot{n}_3 = n_\infty(v_3) - n_3,
\end{aligned}
\tag{27}
$$

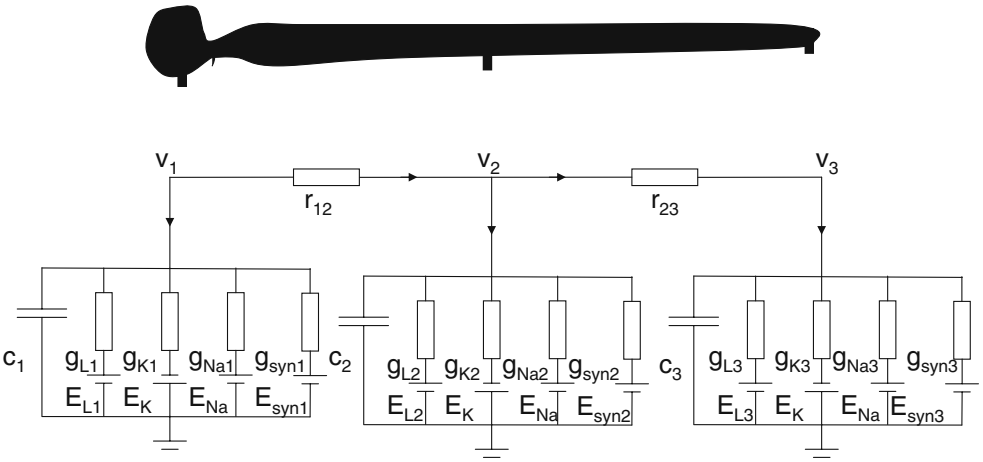

Fig. 8 The equivalent circuit of a neuron with active dendrite consisting of three compartments

where

$$
\begin{aligned}
i_{Nak} &= g_{Nak} m_k^3 h_k (v_k - E_{Na}) \\
i_{Kk} &= g_{Kk} n_k^4 (v_k - E_K) \\
i_{Lk} &= g_{Lk} (v_k - E_{Lk}) \\
i_{synk} &= g_{synk} (E_{synk} - v_k)
\end{aligned} \tag{28}
$$

for $k = 1, 2, 3$.

Let $\bar{v}_k, \bar{m}_k, \bar{h}_k, \bar{n}_k$ be the rest state potential and gating variables. Given a small number ϵ, if $g_{synk} = \epsilon \tilde{g}_{synk}$, the perturbed voltage and gating variables v_k, m_k, h_k, n_k can be approximated by perturbation analysis

$$
\begin{aligned}
v_k &= \bar{v}_k + \epsilon \tilde{v}_k + O(\epsilon^2) \\
m_k &= \bar{m}_k + \epsilon \tilde{m}_k + O(\epsilon^2) \\
h_k &= \bar{h}_k + \epsilon \tilde{h}_k + O(\epsilon^2) \\
n_k &= \bar{n}_k + \epsilon \tilde{n}_k + O(\epsilon^2).
\end{aligned} \tag{29}
$$

Substituting Eq. (29) into Eq. (27), a linearized model about the resting state is obtained by equating the perturbation terms $\tilde{v}_k, \tilde{m}_k, \tilde{h}_k, \tilde{n}_k, \tilde{g}_{synk}$ of order ϵ. The linearized model is described by the following equations

$$
\begin{aligned}
&c_1 \dot{\tilde{v}}_1 + (\tilde{v}_1 - \tilde{v}_2)/r_{12} + g_1 \tilde{v}_1 + i_{m1} \tilde{m}_1 + i_{n1} \tilde{n}_1 + i_{h1} \tilde{h}_1 \\
&\quad = \tilde{i}_{syn1} \\
&c_2 \dot{\tilde{v}}_2 + (\tilde{v}_2 - \tilde{v}_3)/r_{23} - (\tilde{v}_1 - \tilde{v}_2)/r_{12} + g_2 \tilde{v}_2 + i_{m2} \tilde{m}_2 \\
&\quad + i_{n2} \tilde{n}_2 + i_{h2} \tilde{h}_2 = \tilde{i}_{syn2} \\
&c_3 \dot{\tilde{v}}_3 - (\tilde{v}_2 - \tilde{v}_3)/r_{23} + g_3 \tilde{v}_3 + i_{m3} \tilde{m}_3 + i_{n3} \tilde{n}_3 + i_{h3} \tilde{h}_3 \\
&\quad = \tilde{i}_{syn3}
\end{aligned}
$$

$$
\begin{aligned}
\tau_m(\bar{v}_1) \dot{\tilde{m}}_1 &= \dot{m}_\infty(\bar{v}_1) \tilde{v}_1 - \tilde{m}_1 \\
\tau_m(\bar{v}_2) \dot{\tilde{m}}_2 &= \dot{m}_\infty(\bar{v}_2) \tilde{v}_2 - \tilde{m}_2 \\
\tau_m(\bar{v}_3) \dot{\tilde{m}}_3 &= \dot{m}_\infty(\bar{v}_3) \tilde{v}_3 - \tilde{m}_3 \\
\tau_n(\bar{v}_1) \dot{\tilde{n}}_1 &= \dot{n}_\infty(\bar{v}_1) \tilde{v}_1 - \tilde{n}_1 \\
\tau_n(\bar{v}_2) \dot{\tilde{n}}_2 &= \dot{n}_\infty(\bar{v}_2) \tilde{v}_2 - \tilde{n}_2 \\
\tau_n(\bar{v}_3) \dot{\tilde{n}}_3 &= \dot{n}_\infty(\bar{v}_3) \tilde{v}_3 - \tilde{n}_3 \\
\tau_h(\bar{v}_1) \dot{\tilde{h}}_1 &= \dot{h}_\infty(\bar{v}_1) \tilde{v}_1 - \tilde{h}_1 \\
\tau_h(\bar{v}_2) \dot{\tilde{h}}_2 &= \dot{h}_\infty(\bar{v}_2) \tilde{v}_2 - \tilde{h}_2 \\
\tau_h(\bar{v}_3) \dot{\tilde{h}}_3 &= \dot{h}_\infty(\bar{v}_3) \tilde{v}_3 - \tilde{h}_3,
\end{aligned} \tag{30}
$$

where

$$
\begin{aligned}
g_k &= \left(g_{Nak} \bar{m}_k^3 \bar{n}_k + g_{Kk} \bar{h}_k^4 + g_{Lk} \right) \\
i_{mk} &= 3 g_{Nak} \bar{m}_k^2 \bar{n}_k (\bar{v}_k - E_{Na}) \\
i_{nk} &= g_{Nak} \bar{m}_k^3 (\bar{v}_k - E_{Na}) \\
i_{hk} &= 4 g_{Kk} \bar{h}_k^3 (\bar{v}_k - E_K) \\
\tilde{i}_{synk} &= (E_{synk} - \bar{v}_k) \tilde{g}_{synk}
\end{aligned} \tag{31}
$$

for $k = 1, 2, 3$.

It is easy to verify, the above equations can be written into the following block matrix form

$$
\begin{bmatrix} \mathbf{C} & \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{T}_m & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{T}_h & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{T}_n \end{bmatrix} \begin{bmatrix} \dot{\tilde{\mathbf{v}}} \\ \dot{\tilde{\mathbf{m}}} \\ \dot{\tilde{\mathbf{h}}} \\ \dot{\tilde{\mathbf{n}}} \end{bmatrix} = - \begin{bmatrix} \mathbf{G} & \mathbf{I}_m & \mathbf{I}_h & \mathbf{I}_n \\ \dot{\mathbf{M}}_\infty & \mathbf{I} & \mathbf{O} & \mathbf{O} \\ \dot{\mathbf{H}}_\infty & \mathbf{O} & \mathbf{I} & \mathbf{O} \\ \dot{\mathbf{N}}_\infty & \mathbf{O} & \mathbf{O} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{v}} \\ \tilde{\mathbf{m}} \\ \tilde{\mathbf{h}} \\ \tilde{\mathbf{n}} \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \\ \mathbf{O} \\ \mathbf{O} \end{bmatrix} \tilde{\mathbf{i}}_{syn}, \tag{32}
$$

where

$$\tilde{\mathbf{v}} = \begin{bmatrix} \tilde{v}_1 \\ \tilde{v}_2 \\ \tilde{v}_3 \end{bmatrix} \tilde{\mathbf{m}} = \begin{bmatrix} \tilde{m}_1 \\ \tilde{m}_2 \\ \tilde{m}_3 \end{bmatrix} \tilde{\mathbf{h}} = \begin{bmatrix} \tilde{h}_1 \\ \tilde{h}_2 \\ \tilde{h}_3 \end{bmatrix}$$

$$\tilde{\mathbf{n}} = \begin{bmatrix} \tilde{n}_1 \\ \tilde{n}_2 \\ \tilde{n}_3 \end{bmatrix} \tilde{\mathbf{i}}_{syn} = \begin{bmatrix} (E_{syn1} - \bar{v}_1)\tilde{g}_{syn1} \\ (E_{syn2} - \bar{v}_2)\tilde{g}_{syn2} \\ (E_{syn3} - \bar{v}_3)\tilde{g}_{syn3}. \end{bmatrix}. \tag{33}$$

In the above matrix equations, $\mathbf{O}$ denotes a zero matrix block, $\mathbf{I}$ denotes the identity matrix. Denote a diagonal matrix whose diagonal entries are $a_1, \ldots, a_n$ by $diag(a_1, \ldots, a_n)$, we have

$$\mathbf{C} = diag(c_1, c_2, c_3)$$

$$\mathbf{T}_m = diag(\tau_m(\bar{v}_1), \tau_m(\bar{v}_2), \tau_m(\bar{v}_3))$$

$$\mathbf{T}_h = diag(\tau_h(\bar{v}_1), \tau_h(\bar{v}_2), \tau_h(\bar{v}_3))$$

$$\mathbf{T}_n = diag(\tau_n(\bar{v}_1), \tau_n(\bar{v}_2), \tau_n(\bar{v}_3))$$

$$\mathbf{I}_m = diag(i_{m1}, i_{m2}, i_{m3})$$

$$\mathbf{I}_h = diag(i_{h1}, i_{h2}, i_{h3})$$

$$\mathbf{I}_n = diag(i_{n1}, i_{n2}, i_{n3})$$

$$\dot{\mathbf{M}}_\infty = diag(\dot{m}_\infty(\bar{v}_1), \dot{m}_\infty(\bar{v}_2), \dot{m}_\infty(\bar{v}_3))$$

$$\dot{\mathbf{H}}_\infty = diag(\dot{h}_\infty(\bar{v}_1), \dot{h}_\infty(\bar{v}_2), \dot{h}_\infty(\bar{v}_3))$$

$$\dot{\mathbf{N}}_\infty = diag(\dot{n}_\infty(\bar{v}_1), \dot{n}_\infty(\bar{v}_2), \dot{n}_\infty(\bar{v}_3))$$

$$\mathbf{B} = diag(1, 1, 1)$$

$$\mathbf{G} = \begin{bmatrix} g_1 + 1/r_{12} & -1/r_{12} & 0 \\ -1/r_{12} & g_2 + 1/r_{12} + 1/23 & -1/r_{23} \\ 0 & -1/r_{23} & g_3 + 1/r_{23} \end{bmatrix}. \tag{34}$$

In the quasi-active model, we are often interested in the change of somatic potential under synaptic stimulation. If the voltage potential at node k, $\tilde{v}_k (k \leq 3)$, is the somatic potential, it can be obtained by

$$\tilde{v}_k = \begin{bmatrix} \mathbf{e}_k & \mathbf{o} & \mathbf{o} & \mathbf{o} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{v}} \\ \tilde{\mathbf{m}} \\ \tilde{\mathbf{h}} \\ \tilde{\mathbf{n}} \end{bmatrix} = \mathbf{e}_k \tilde{\mathbf{v}}, \tag{35}$$

where $\mathbf{e}_k$ is a row vector whose kth element is 1 and all the other elements are 0, and $\mathbf{o}$ denotes zero row vector.

If we denote

$$\mathbf{x} = \begin{bmatrix} \tilde{\mathbf{v}} \\ \tilde{\mathbf{m}} \\ \tilde{\mathbf{h}} \\ \tilde{\mathbf{n}} \end{bmatrix}, \mathcal{C} = \begin{bmatrix} \mathbf{C} & \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{T}_m & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{T}_h & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{T}_n \end{bmatrix},$$

$$\mathcal{G} = \begin{bmatrix} \mathbf{G} & \mathbf{I}_m & \mathbf{I}_h & \mathbf{I}_n \\ \dot{\mathbf{M}}_\infty & \mathbf{I} & \mathbf{O} & \mathbf{O} \\ \dot{\mathbf{H}}_\infty & \mathbf{O} & \mathbf{I} & \mathbf{O} \\ \dot{\mathbf{N}}_\infty & \mathbf{O} & \mathbf{O} & \mathbf{I} \end{bmatrix}, \mathcal{B} = \begin{bmatrix} \mathbf{B} \\ \mathbf{O} \\ \mathbf{O} \\ \mathbf{O} \end{bmatrix}, \mathbf{l} = \begin{bmatrix} \mathbf{e}_k & \mathbf{o} & \mathbf{o} & \mathbf{o} \end{bmatrix}, \tag{36}$$

Eqs. (32) and (35) can be written into the following state-space form

$$\begin{aligned} \mathcal{C}\dot{\mathbf{x}} &= -\mathcal{G}\mathbf{x} + \mathcal{B}\tilde{\mathbf{i}}_{syn} \\ \tilde{v}_k &= \mathbf{l}^T \mathbf{x}. \end{aligned} \tag{37}$$

In the above example, there are three compartments, three gating variables for each compartment, and three synaptic inputs. However, it is easy to see the block formulation (36) can be extended to more general cases.

Given a neuron model with n compartments, q gating variables for each compartment, and p synaptic inputs, the order of the state-space model (37) will be $N = (q + 1)n$ and the dimension of the matrices will be $\mathcal{C} \in \mathbb{R}^{N \times N}, \mathcal{G} \in \mathbb{R}^{N \times N}, \mathcal{B} \in \mathbb{R}^{N \times p}, \mathbf{l} \in \mathbb{R}^{1 \times N}$.

2.3.2 Output Krylov subspace method

Different from the multi-input multi-output (MIMO) system (9), which we deal with in Algorithms 1 and 2, the quasi-active model (37) is multi-input single-output (MISO). For MIMO systems, the efficiency of model reduction will degrade as the number of ports increases. However, for the MISO case, the problem can be significantly simplified. Here, we propose an efficient model reduction method based on output Krylov subspace projection.

Given the state-space representation (37), there are two Krylov subspaces $K_m(\mathcal{G}^{-1}\mathcal{C}, \mathcal{G}^{-1}\mathcal{B})$ and $K_m(\mathcal{G}^{-T}\mathcal{C}^T, \mathcal{G}^{-T}\mathbf{l}^T)$, which are called input and output Krylov subspaces, respectively. It can be proved that if projection matrix $\mathbf{V}$ is a basis of the input Krylov subspace or output Krylov subspace, then the reduced model will match first m moments (Villemagne and Skelton 1987; Salimbahrami and Lohmann 2002).

Assume the quasi-active model has p inputs, in order to match m moments, if the input Krylov subspace is used, the reduced order will be pm, which could be large if p is large. However, as the quasi-active model has only one output, if the output Krylov subspace is

used, the reduced order will be m, which is independent of the number of inputs.

Therefore, in the proposed output Krylov subspace method, a reduced model of order m can be obtained by projecting the original equations (37) in the column space of $\mathbf{V}$

$$span(\mathbf{V}) = K_m(\mathcal{G}^{-T}\mathcal{C}^T, \mathcal{G}^{-T}\mathbf{l}^T). \tag{38}$$

The algorithm is given in Algorithm 3. Again, the projection matrix $\mathbf{V}$ is obtained by a numerically efficient Arnoldi method. Note that, as the reduced model is always very small, there is no need to diagonalize reduced matrices $\tilde{\mathcal{C}}$ and $\tilde{\mathcal{G}}$ here.

Algorithm 3 Model order reduction of quasi-active systems

Input: $\mathbf{Z}(s) : (\mathcal{G}, \mathcal{C}, \mathcal{B}, \mathbf{l}), m$
Output: $\tilde{\mathbf{Z}}(s) : (\tilde{\mathcal{C}}, \tilde{\mathcal{G}}, \tilde{\mathcal{B}}, \tilde{\mathbf{l}})$

1. Solve $\mathcal{G}^T\mathbf{r} = \mathbf{l}^T$ for $\mathbf{r}$
2. Normalize $\mathbf{v}_0 = \mathbf{r}/||\mathbf{r}||$
3. For $k = 1, \ldots, m$
   Set $\mathbf{v} = \mathcal{C}^T\mathbf{x}_{k-1}$
   Solve $\mathcal{G}^T\mathbf{x}_k^{(0)} = \mathbf{v}$ for $\mathbf{x}_k^{(0)}$
   For $j = 1, \ldots, k$
      $H = \mathbf{x}_{k-j}^T\mathbf{x}_k^{(j-1)}$
      $\mathbf{x}_k^{(j)} = \mathbf{x}_k^{(j-1)} - \mathbf{x}_{k-j}H$
   Normalize $\mathbf{v}_k = \mathbf{r}/\left|\left|\mathbf{v}^{(k)}\right|\right|$
4. Set $\mathbf{V} = [\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_{k-1}]$
5. Compute $\tilde{\mathcal{C}} = \mathbf{V}^T\mathcal{C}\mathbf{V}, \tilde{\mathcal{G}} = \mathbf{V}^T\mathcal{G}\mathbf{V}, \tilde{\mathcal{B}} = \mathbf{V}^T\mathcal{B}, \tilde{\mathbf{l}} = \mathbf{l}\mathbf{V}$

2.4 Model parameters and simulation protocols

To demonstrate the effectiveness of the proposed methods, we use the passive dendrite of a layer V pyramidal neuron from the rat barrel cortex (Dyhrfjeld-Johnsen et al. 2005) as an example. The morphology is available online in the database of Neuron simulator (NeuronDB). The passive dendrite consists of 2059 coupled cylindrical compartments with resistive and capacitive elements. The resting potential is $-0.07V$ and the values of the passive parameters (specific membrane resistance R_m, specific membrane capacitance C_m, and axial resistivity R_i) are $C_m = 0.007F/m^2$, $R_m = 4\Omega.m^2$, and $R_i = 0.9\Omega.m$.

In the following section, we present the results of simulations based on reduced passive and quasi-active dendrite models, respectively. All the simulations are performed with an Intel Duo Core CPU with 3.17 GHz. The codes are written in Matlab 7.0 (http://dropzone.tamu.edu/~byan).

2.4.1 Passive reduced model simulation

In Section 3.1, to assess the accuracy of the reduced model for passive dendrite, we perform spectrum simulation under white noise inputs, somatic current injection, and synaptic integration, in Sections 3.1.1, 3.1.2, and 3.1.3, respectively. The original passive dendrite is described by 2059 differential equations and has an order of 2059. 100 ports are considered and the reduced passive dendrite has an order 54. The reduced dendrite is generated by Algorithm 2 with $m = 5$ and $p = 0.5$.

For somatic current injection and synaptic integration, active conductances are placed at the soma to generate spiking trains of action potentials. The conductance follows Hodgkin–Huxley-like kinetics (Hodgkin and Huxley 1952) with fast sodium Na and delayed rectifier potassium K. The current-voltage relationship of somatic ion channels are given in Eq. (1). For the fast Na, the maximum conductance and reversal potential are 4,800 S/m^2 and 0.05 V. For the delayed rectifier potassium, the maximum conductance and reversal potential are 1,250 S/m^2 and -0.09 V. The synaptic inputs are modeled by a time-varying conductance $g(t)$ in series with a battery with reversal potential of 0 V. The conductance is modeled by randomly generated trains of alpha functions (Rall 1967; Jack et al. 1975) and each alpha function has a peak amplitude of 1 nS and a time to peak of 1 ms.

2.4.2 Quasi-active reduced model simulation

In Section 3.2, spectrum simulation under white noise inputs, somatic current injection, and synaptic integration are performed in Sections 3.2.1, 3.2.2, and 3.2.3, respectively. The quasi-active model is also generated based on the morphology of the previous pyramidal cell with 2,059 compartments. In this quasi-active example, Na and K channels with Hodgkin–Huxley kinetics are uniformly distributed in each compartment. After linearization as described in Section 2.3, we obtain a state-space model (37) of order 8236. In this example, 100 ports are considered and Algorithm 3 is applied to generate a reduced model of order 5.

2.4.3 A simple network simulation

The simulations of passive reduced dendrites in Section 3.1 are based on a single cell. To demonstrate the network behavior of the reduced model, we use pyramidal neurons as units to construct a network in Section 3.3. The network is illustrative and very simple. The aim of this simulation is not to provide a very realistic neocortical model but to show that the reduced

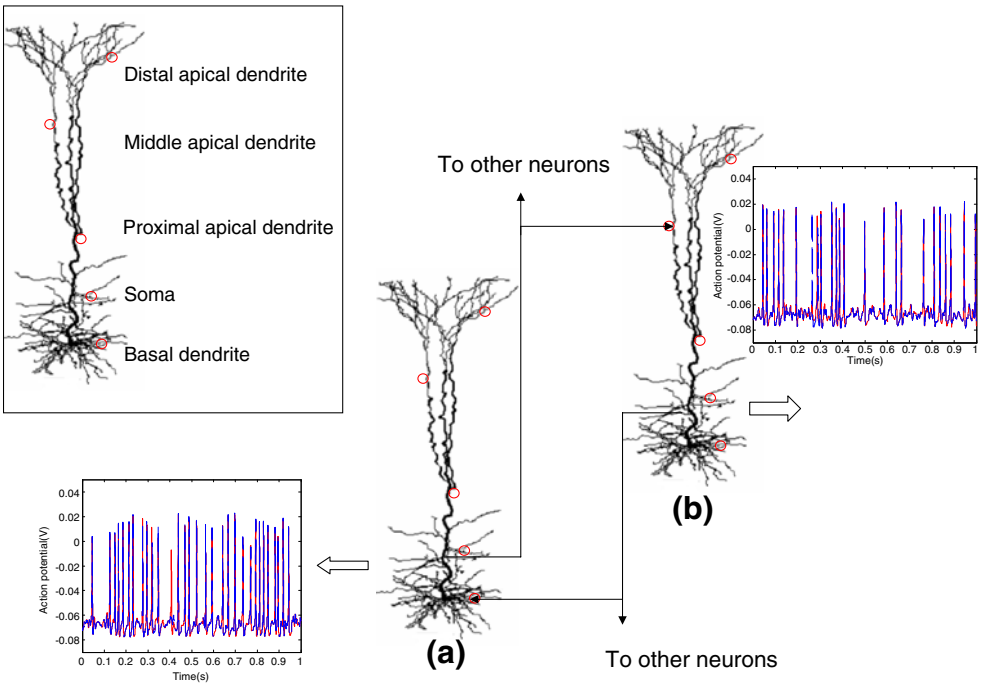

Fig. 9 A simple network simulation

model can preserve the output waveforms of all the cells in a network level simulation.

Compared with the previous single cell simulation, in this network example, we consider a much simpler representation of the pyramidal cell. As shown in Fig. 9, the dendrite tree is divided into five regions: distal apical dendrite, middle apical dendrite, proximal apical dendrite, soma, and basal. In each region, one port is randomly selected.

There are ten pyramidal cells in the network. As shown in Fig. 9, each pyramidal cell receives excitatory synaptic inputs from other cells through one of the ports and passes spikes to other cells through the axon. In this example, we regard each axon as a simple delay line for the propagation of spikes (Bower and Beeman 1998). In addition, those cells also receive synaptic inputs modeled as randomly distributed trains of spikes, which represent the random spontaneous "background level" firing from many other neurons not being modeled in the network.

3 Results

3.1 Passive reduced model simulation

3.1.1 Spectrum simulation under white noise inputs

As shown in Fig. 10, given the passive dendrite, we first connect independent current sources at each port and the randomly fluctuating input currents are modeled as Gaussian white noise. The voltage response at Port 1 corresponds to the somatic potential.

The voltage responses generated by original and reduced models are compared in Fig. 10. It is easy to see the time domain responses are indistinguishable. The corresponding power spectra are also shown in Fig. 10. As the dendrite is modeled as an RC circuit, which behaves as low-pass filters, the power spectrum decays dramatically as the frequency increases. The fact that power spectra of both models are indistinguishable at lower frequencies implies the indistinguishable responses in the time domain.

3.1.2 Somatic current injection

Now we perform simulations under somatic negative and positive current injections. As shown in Fig 11, active conductances are placed at the soma (Port 1) to generate spiking trains of action potentials.

Somatic negative current injection is a classical method of identifying two important parameters of a passive dendrite tree: somatic input resistance R_{in} and time constant τ_m (Rall 1964; Bush and Sejnowski 1993). The responses of the reduced and full models to a constant somatic current injection of -0.7 nA are compared in Fig. 11(a). The overlapping of the two curves shows that both models have the same R_{in} and τ_m.

In addition, we perform simulations to observe the firing responses of the cell to a somatic positive current

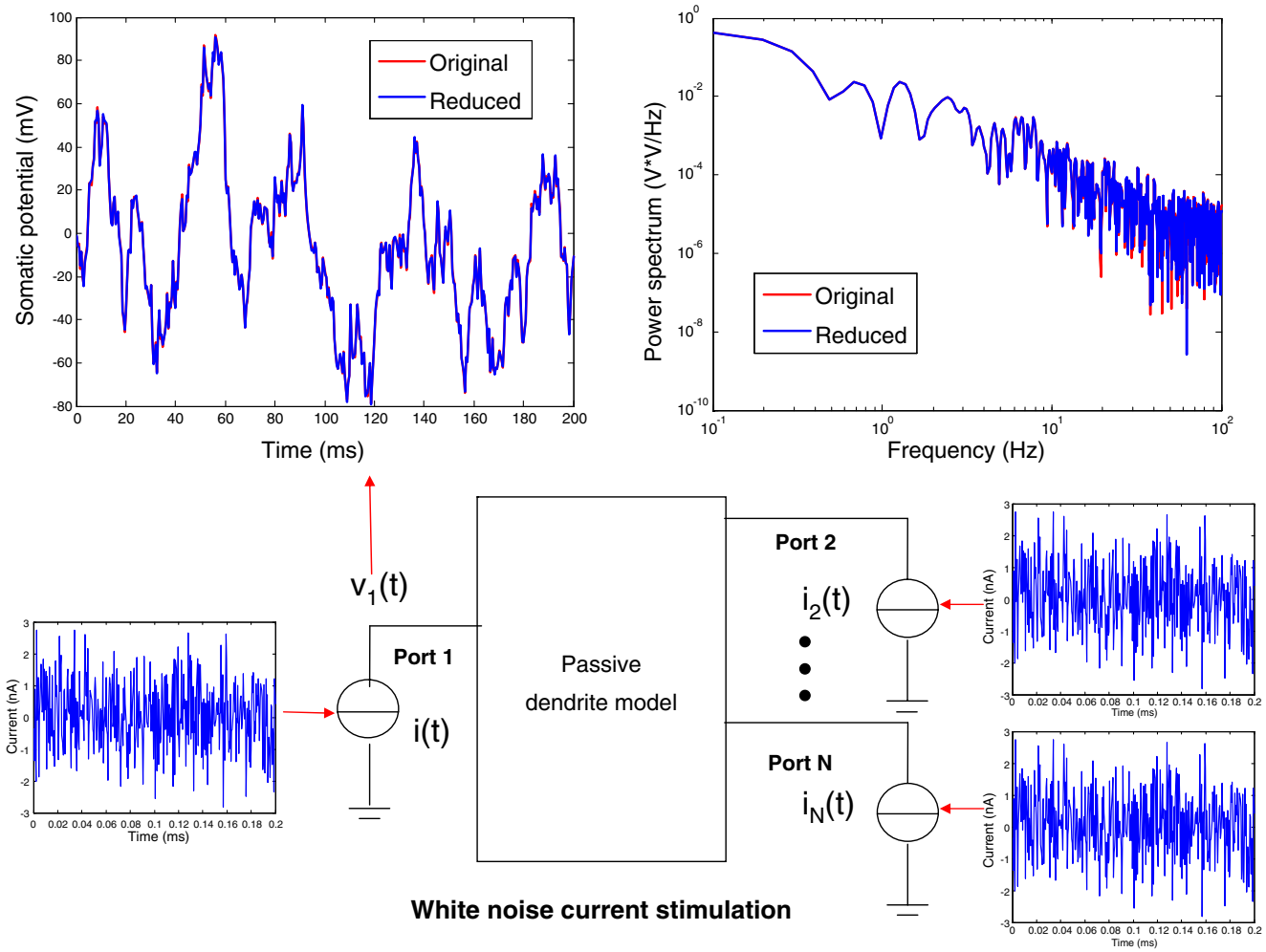

Fig. 10 Simulation of passive dendrites under white noise input currents

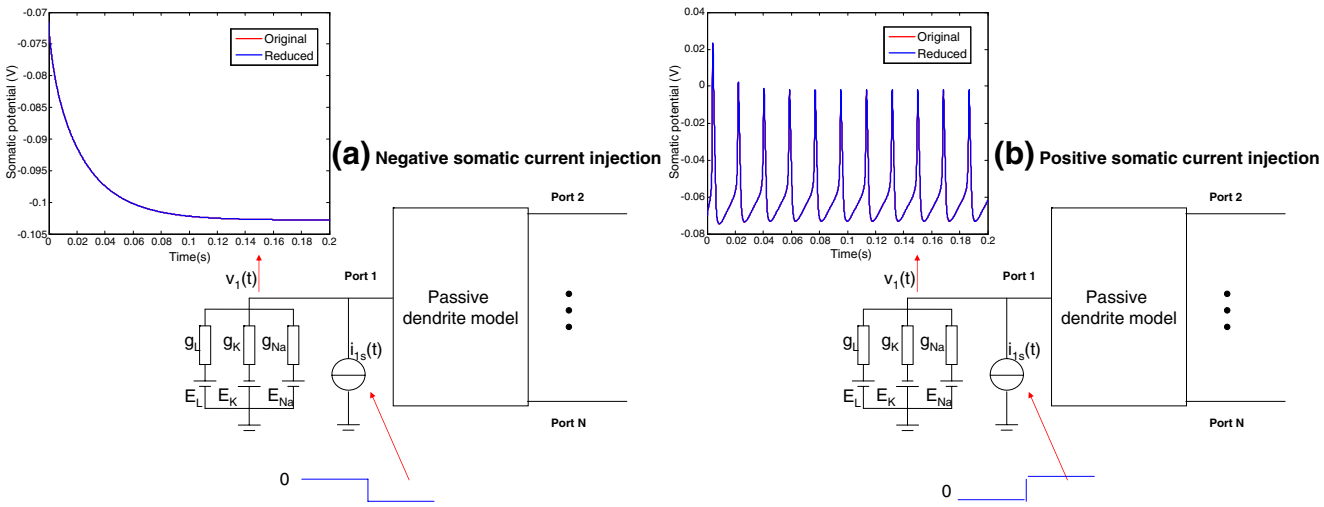

Fig. 11 Simulations of somatic current injections.

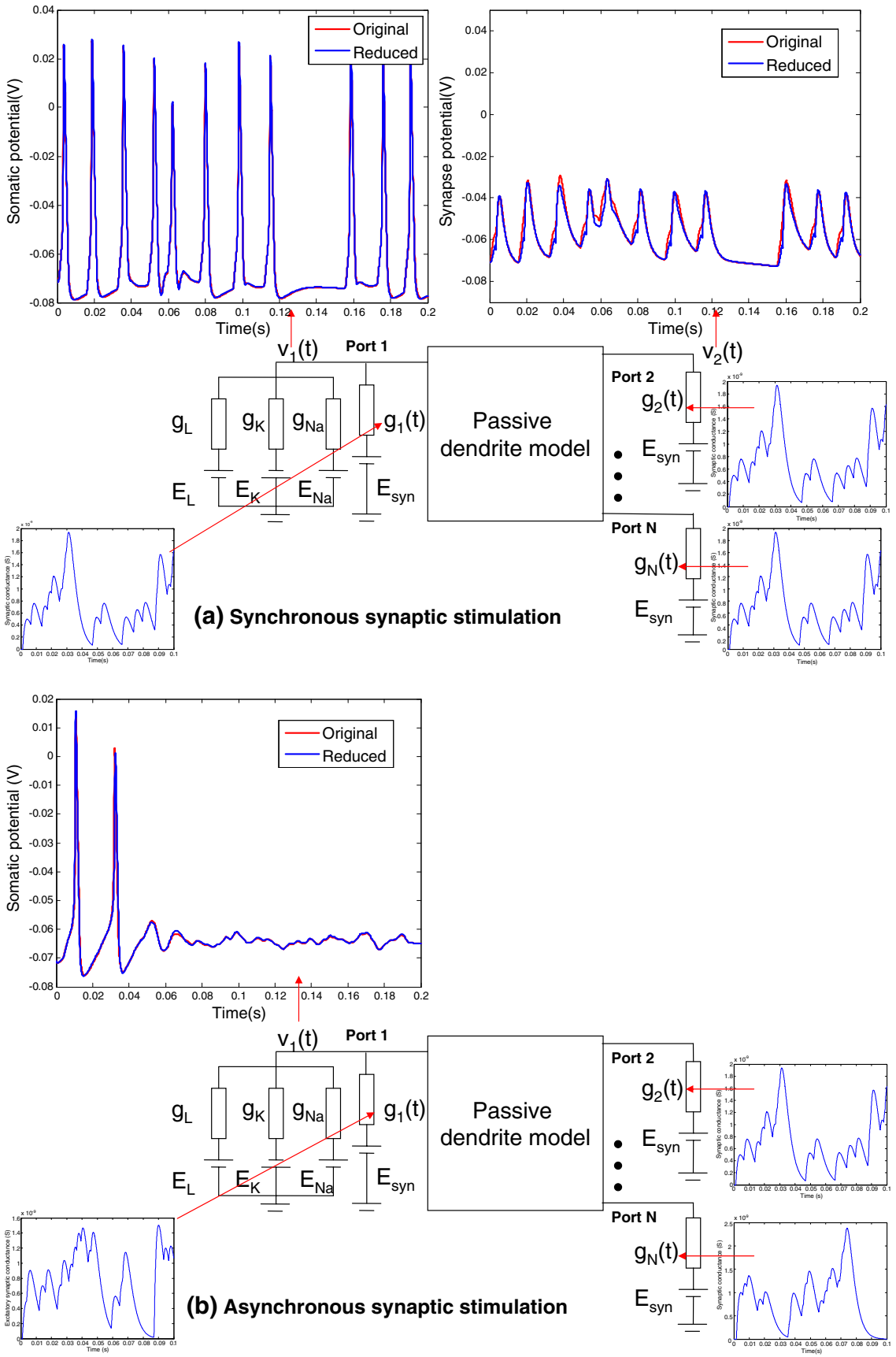

Fig. 12 Simulations of synaptic integrations

injection. The spiking responses of the reduced and full models to a maintained 1.5 nA somatic current injection are given in Fig. 11(b), where the two responses are indistinguishable.

3.1.3 Synaptic integration

A fundamental function of nerve cells is the synaptic integration, which transforms the incoming synaptic information into specific patterns of action potential output. To demonstrate how accurately the reduced model preserves the synaptic integration properties of the full model, as illustrated in Fig. 12, we perform simulations to compare the responses of both models to the same synaptic inputs.

Figure 12(a) and (b) shows the responses of both models to synchronous and asynchronous synaptic inputs from 100 different locations. It is easy to see the neuron tends to fire more easily when receiving synchronous synaptic inputs. In both cases, the reduced model can reproduce exactly the same action potential waveforms of the full model. Another thing worth mentioning is that, in addition to the somatic response (Port 1), the reduced model also preserves the voltage responses at other ports to a certain degree. For example, the voltage responses at a dendrite location (Port 2) are also given in Fig. 12(a). As shown in Fig. 12(a), the signal at the dendritic location (Port 2) often has less high frequency components than the signal at somatic location (Port 1), which is one of the reasons we could match less moments for dendritic inputs.

We quantify the performance of reduced models in terms of speedup and accuracy as the number of ports increases. Given the number of ports, Algorithm 2 ($m = 5$, $p = 0.5$) generates reduced models of different orders automatically. In each case, the models are stimulated by synchronous synaptic inputs as in Fig. 12(a). From Table 1, we can see the speedup decreases from 25.27X to 3.35X as the number of ports increases from 1 to 100. The timing accuracy is measured by the mean displacement of spikes defined by $d = \sum_{i=1}^{K} t_k/K$, where t_k is the time of the kth spike and K is the total number of spikes. If d and d_r denote the mean spike displacements for original and reduced models respectively, the error is measured as $|d - d_r|$.

In Table 1, most of the errors are less than 0.1 ms, which means the reduced models can reproduce the spiking trains of the original models very accurately. Figure 12(a) corresponds to the case with 100 ports (last row in Table 1). The reason why reduced models with 10 and 20 ports are less accurate (>0.1 ms) is that the current configuration of $p(0 \le p \le 1)$ in Algorithm 2 is in favor of models with a large number of ports. The configuration $p = 0.5$ means only 50% non-somatic ports are included in the projection matrix. This is to ensure the compactness of the reduced models with a large number of ports. In fact, for models with a small number of ports, the accuracy can be simply improved by choosing a larger p and this will be demonstrated later in Table 2.

Table 1 The performance of Algorithm 2 ($m = 5$, $p = 0.5$)

Number of port	Reduced order	Runtime			Timing error $\lvert d - d_r \rvert$ (ms)
		Original (s)	Reduced (s)	Speedup (X)	
1	5	12.13	0.48	25.27	0.0045
10	9	12.40	0.80	15.50	3.2045
20	14	12.67	1.13	11.31	0.1455
30	19	13.24	1.63	8.12	0.0318
40	24	13.80	2.08	6.63	0.0727
50	29	13.90	2.51	5.54	0.0409
60	34	13.95	2.91	4.79	0.0318
70	39	14.50	3.26	4.45	0.0091
80	44	14.54	3.70	3.93	0.0227
90	49	14.85	3.99	3.72	0.0091
100	54	15.23	4.55	3.35	0.0182

As the proposed methods do not take advantage of topological information of the dendritic tree, the selection of port locations has little effect. To illustrate this point, we perform experiments with 10 and 100 ports, respectively. In both cases, five moments are matched for the somatic port ($m = 5$). In the former case, all the non-somatic ports are included in the projection matrix ($p = 1$) and the reduced order is 14. In the latter case, 50% non-somatic ports are included in the projection matrix ($p = 0.5$) and the reduced order is 54 (same as Table 1).

For both cases, 20 simulations are performed. In each simulation, the locations of the ports are randomly distributed over the full dendritic tree. As shown in Table 2, in both cases, the maximum timing errors are less than 0.1 ms and the variance of performance is small. This means the proposed methods are very accurate and the selection of port locations has little effect.

Table 2 The performance of Algorithm 2 with different selections of port locations

Number of port	Reduced order	Timing error $\lvert d - d_r \rvert$			
		Min (ms)	Max (ms)	Mean (ms)	Standard deviation (ms)
10	14	0.0100	0.0838	0.0495	0.0266
100	54	0.0091	0.0636	0.0343	0.0125

3.2 Quasi-active reduced model simulation

3.2.1 Spectrum simulation under white noise inputs

As shown in Fig. 13, given the quasi-active model, we add independent current sources at each port and the randomly fluctuating input synaptic currents are modeled as Gaussian white noise. The voltage responses at Port 1 corresponds to the somatic potential. The voltage responses generated by original and reduced models and the corresponding power spectra are compared in Fig. 13. Similar to the passive dendrite case, the quasi-active system also behaves as low-pass filters, i.e., the power spectrum decays dramatically as the frequency increases. As the power spectra of both models are indistinguishable at lower frequencies, where the most power is concentrated, the time domain responses are indistinguishable.

3.2.2 Somatic current injection

Now we perform simulations under somatic negative and positive current injections. As shown in Fig. 14, the responses of the reduced and full models to a constant somatic current injection of 0.1 nA and −0.1 nA are compared in Fig. 14(a) and (b). In both cases, the two responses are indistinguishable.

3.2.3 Synaptic integration

To demonstrate how accurately the reduced model preserves the synaptic integration properties of the full model, as illustrated in Fig. 15, we connect independent current sources at each port. Note that, as the resting potential $\bar{V}_k$ is constant, the synaptic current input at

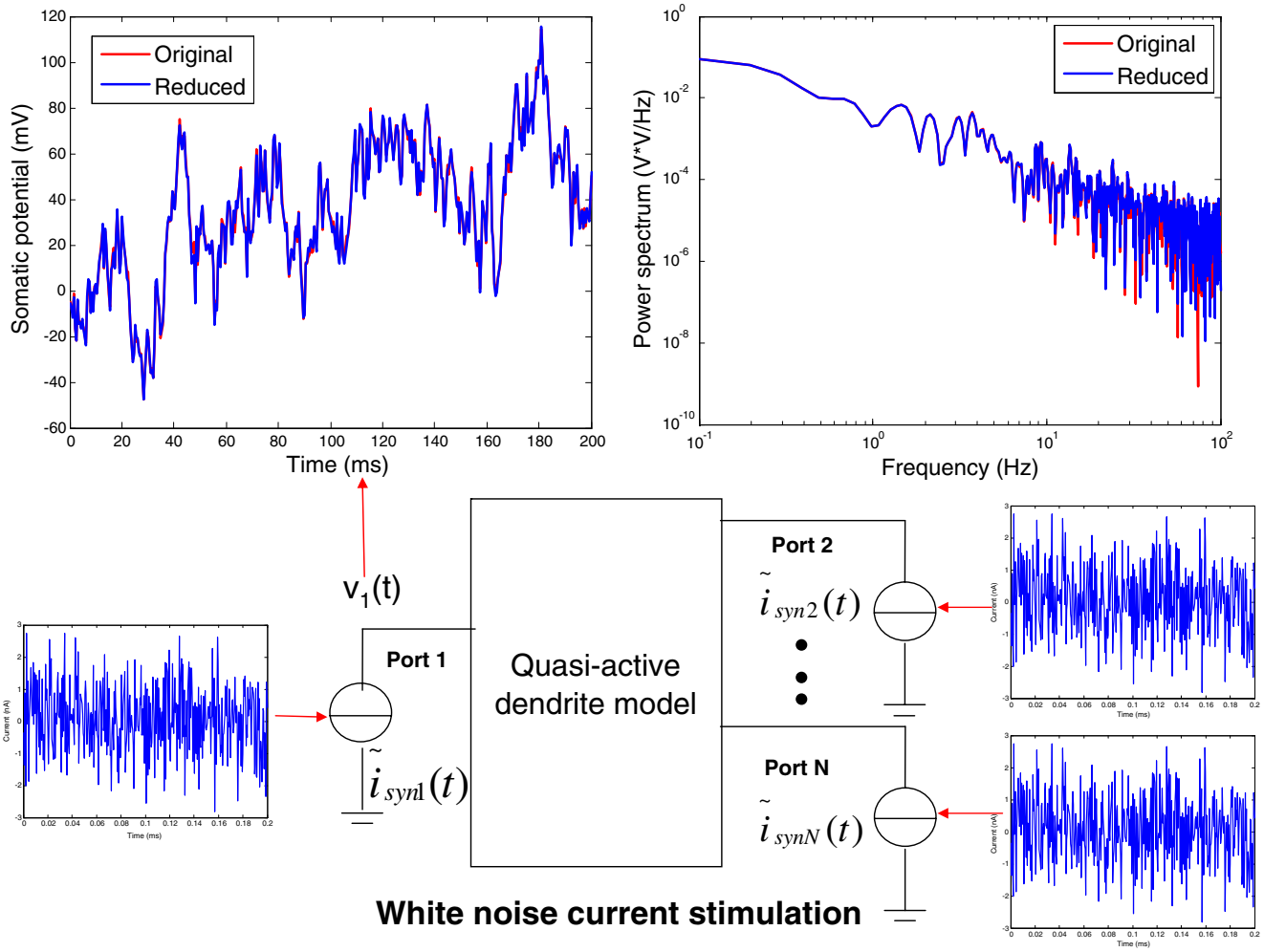

Fig. 13 Simulation of quasi-active models under white noise input currents

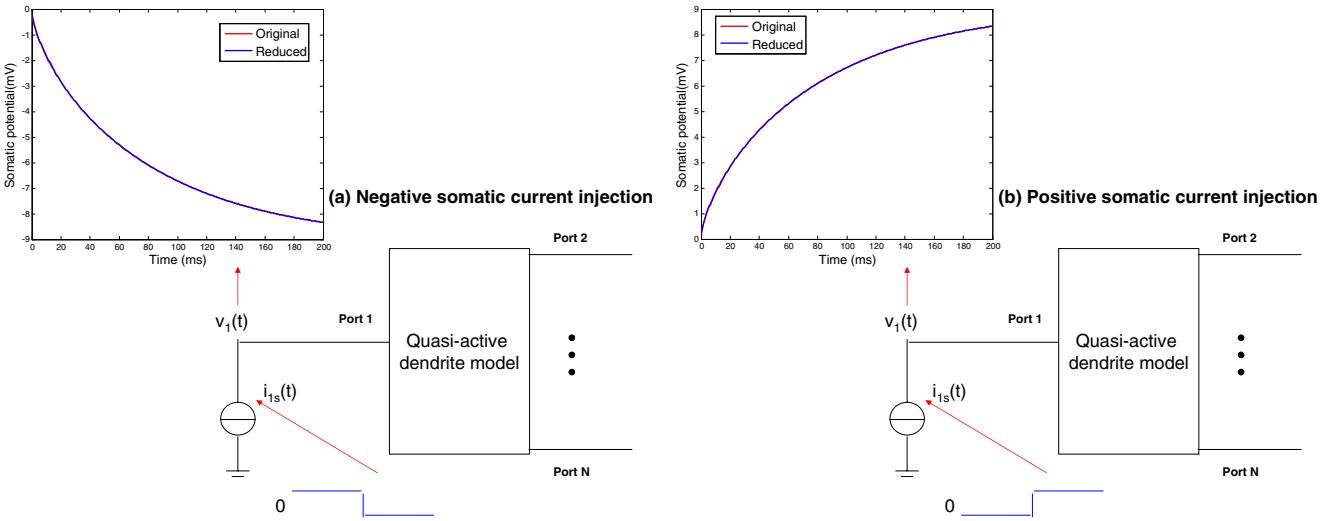

Fig. 14 Simulations of somatic current injections of quasi-active model

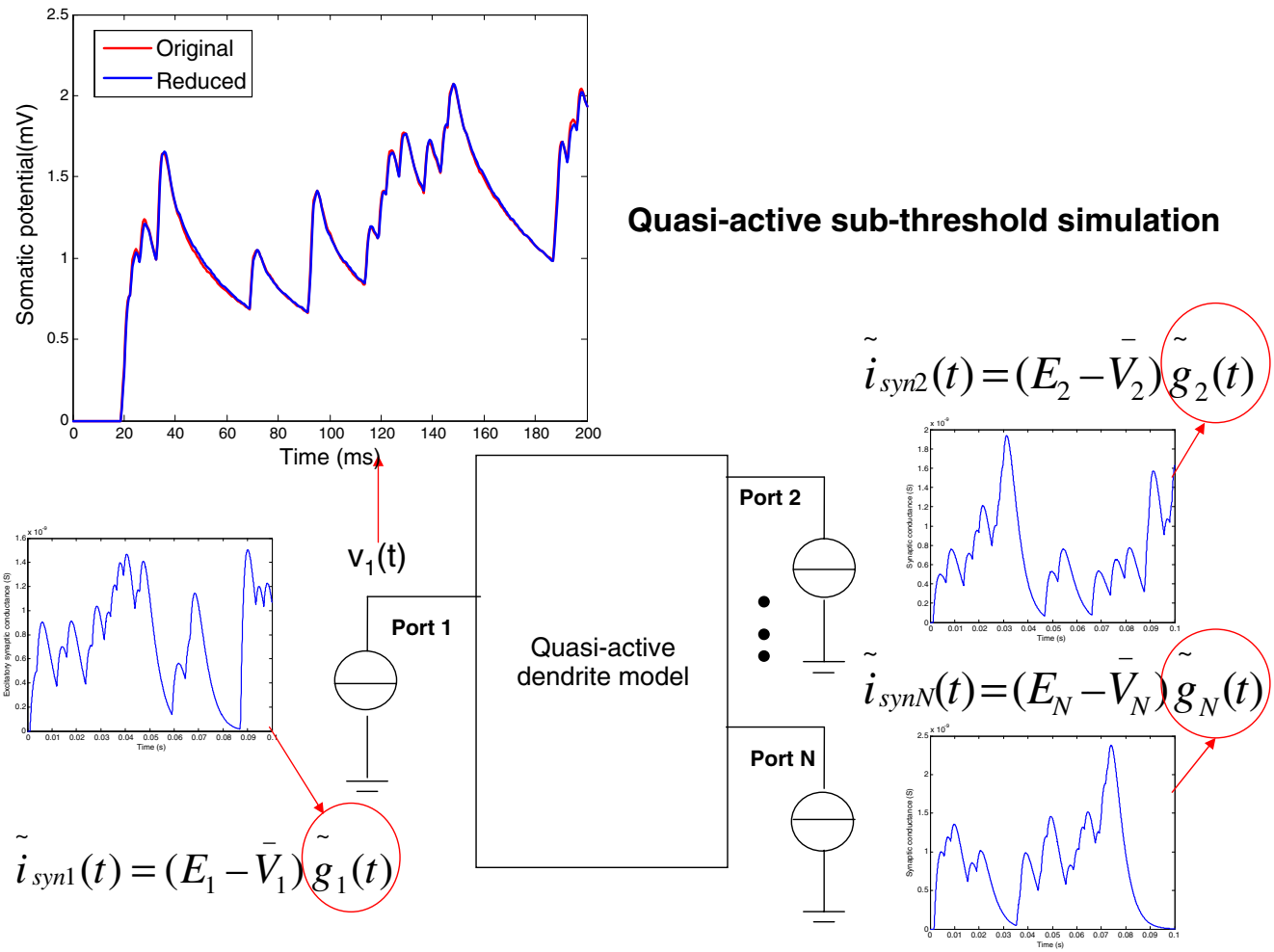

Fig. 15 Simulations of synaptic integrations

the kth port, $\tilde{i}_{synk}(t)$, is proportional to the change of conductance $\tilde{g}_k(t)$

$$\tilde{i}_{synk}(t) = (E_k - \bar{V}_k)\tilde{g}_k(t), \tag{39}$$

and $\tilde{g}_k(t)$ is modeled by randomly generated trains of alpha functions with a peak amplitude of 1 nS and a time to peak of 1 ms. We perform simulations to compare the somatic subthreshold responses of both models to the same synaptic inputs. Figure 15 shows the responses of both models to synaptic inputs from 100 different locations. It is easy to see the reduced model can reproduce the exactly the same waveforms as subthreshold potentials.

We quantify the performance of reduced models from Algorithm 3 in terms of speedup and accuracy. First, we keep the number of ports a constant number, 100, and demonstrate the accuracy of the reduced models in terms of the reduced order. We compare the somatic subthreshold membrane potentials of the original and reduced models at different reduced orders. As there is no firing behavior, we use four different measures to represent the errors: maximum absolute error $|v(t_k) - v_r(t_k)|_{\max}$, mean absolute error $|v(t_k) - v_r(t_k)|_{\text{mean}}$, max relative error $|(v(t_k) - v_r(t_k))/v(t_k)|_{\max}$, and mean relative error $|(v(t_k) - v_r(t_k))/v(t_k)|_{\text{mean}}$. As shown in Table 3, the errors generated by reduced models are very small. Usually, a reduced order of 3 is good enough and there is no need to further increase the order. The table also lists the reduction time, i.e., the extra time needed to build the reduced model. As the reduced order increases, the reduction cost increases slightly and the cost is very low.

Second, we keep the reduced order a constant number, 3, and demonstrate the performance of the reduced models in terms of the number of inputs. As shown in Table 4, with fixed reduced order, the accuracy and reduction time do not change much. However, as the number of inputs increases from 1 to 500, the speedup decreases from 646.8 to 14.6. The reason is that while the $\mathcal{B}$ matrix in original system (37) is sparse, the matrix in the reduced system is dense. For example, given a number of 500 inputs, although the $\mathcal{B}$ in original system has the dimension of 8,236 × 500, there are only 500 nonzero elements. In this case, even if the reduced order is 3, the matrix in the reduced system is a 3 × 500 dense matrix and thus the number of nonzero elements are 1,500.

Table 3 The performance of reduced models of different orders

Reduced order	Reduction time (s)	Absolute error		Relative error	
		Max (mv)	Mean (mv)	Max (%)	Mean (%)
1	0.041	0.345	0.153	0.493	0.218
2	0.061	0.199	0.049	0.286	0.070
3	0.081	0.075	0.013	0.108	0.019
4	0.105	0.032	0.008	0.046	0.011
5	0.122	0.017	0.003	0.024	0.005

Table 4 The performance of reduced models with different number of inputs

Number of inputs	Reduction time (s)	Runtime			Relative error	
		Original (s)	Reduced (s)	Speedup (X)	Max (%)	Mean (%)
1	0.067	7.115	0.011	646.8	0.112	0.016
100	0.067	7.138	0.026	274.5	0.129	0.019
200	0.068	7.147	0.050	142.9	0.090	0.017
300	0.067	7.279	0.194	37.5	0.073	0.017
400	0.067	7.530	0.335	22.5	0.086	0.018
500	0.068	7.627	0.522	14.6	0.076	0.015

3.3 A simple network simulation

The reduced models have an order of 9, which are obtained by Algorithm 2 with $m = 5$ and $p = 1$ (matching five moments of the original model at somatic port and one moment at other ports). Given a longer period of 1 s, the network with reduced models runs about eight times faster than the network with full models and can produce almost the same action potential output waveforms of all the cells. Table 5 provides the performance measure of each reduced neuron model in terms of both correlation coefficient $corr(v(t), v_r(t))$ and the number of firings. If original and reduced models fires N and

Table 5 The performance of reduced models in the network example

Neuron ID	Correlation coefficient (%)	The number of firing		
		Original	Reduced	Accuracy (%)
1	95.75	31	30	96.77
2	93.92	23	24	95.65
3	88.66	34	34	100.00
4	94.22	27	27	100.00
5	91.52	31	29	96.77
6	91.11	34	34	100.00
7	88.96	35	37	94.29
8	96.14	32	31	96.88
9	92.21	37	37	100.00
10	94.61	35	35	100.00

N_r times, respectively, the accuracy measure in terms of the number of firing is given by $|1 - N_r/N|$.

The waveforms of neurons 1–4 are given Fig. 16. For neuron 1(a), the reduced model missed one firing at about 0.4 s. For neuron 2(b), the reduced model fired an additional one at about 0.3 s. For neuron 3(c), although the accuracy in the table is 100%, the reduced model actually missed one firing at about 0.35 s but fired an additional one at about 0.25 s. For neuron 4(d), the reduced model accurately tracks all the firings of the original model. From Fig. 16, we can see, the detailed timing information can be preserved very accurately as well as the firing rate (in spite of occasional mistakes). Note that, although 1 s is used here, similar results can be observed even if simulation period is further increased.

4 Discussion

4.1 Model reduction for neuron modelings

Model reduction is a technique to reduce the internal complexity of a system while preserving the input-output behavior. The technique is well developed and extremely powerful for a class of linear time invariant (LTI) systems. In this case, the simulation cost can be significantly reduced and the input-output behavior of system can be well preserved.

As shown in Fig. 17, the neuron model is a strong nonlinear system due to the nonlinear behavior of active channels. The inputs are reflected by synaptic conductances $g_1(t), g_2(t), \ldots, g_N(t)$ and the output is the somatic voltage response $v_1(t)$. Although techniques of

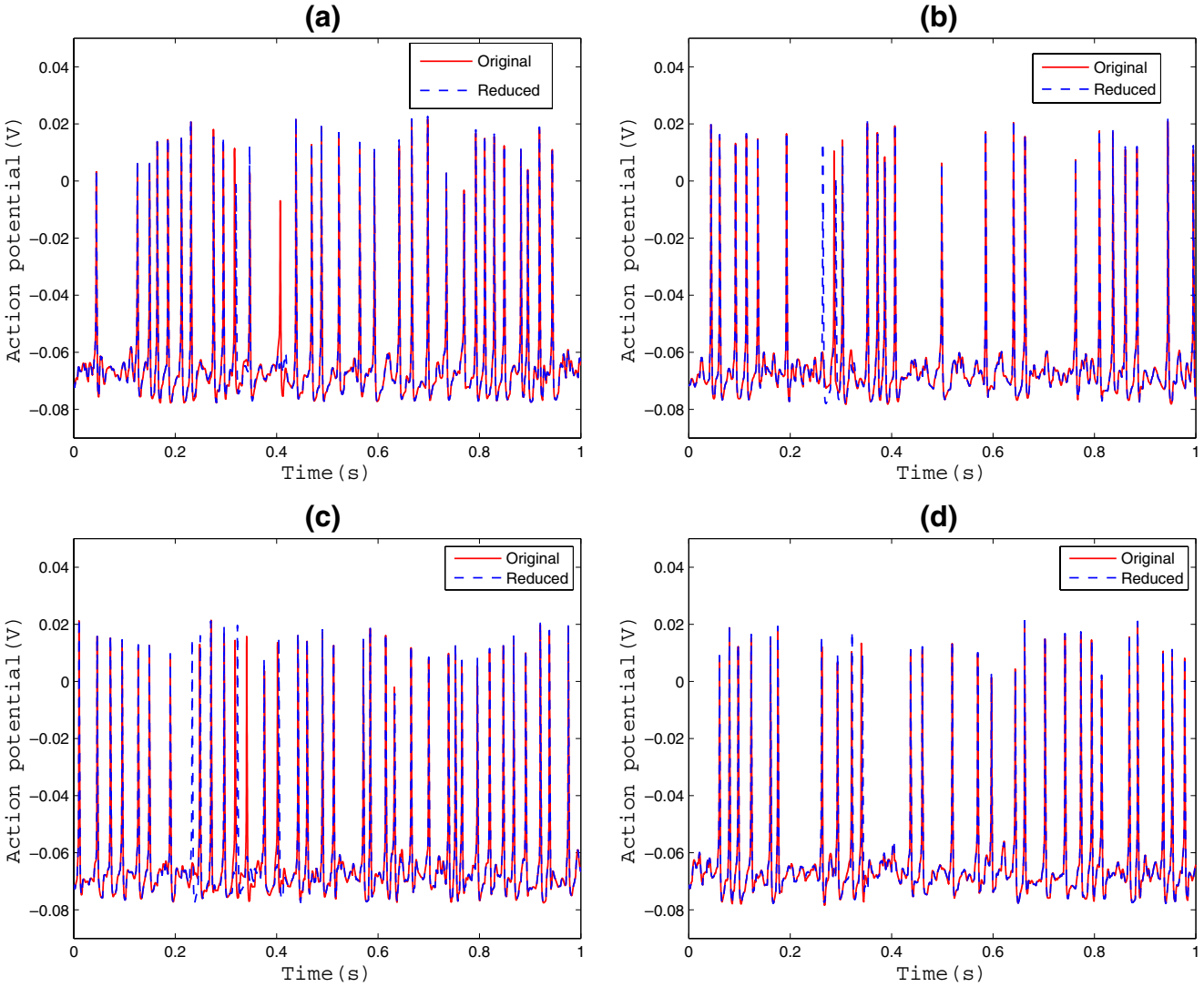

Fig. 16 Firing responses at some of the cells in the network

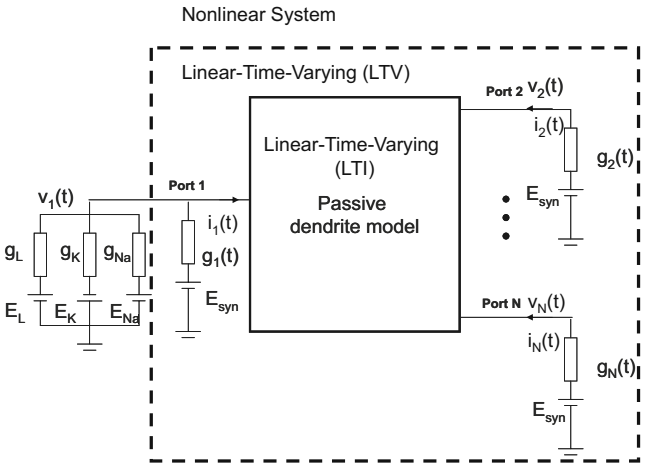

Fig. 17 The neuron model as a dynamic system

model order reduction (MOR) are well developed and extremely powerful in the case of LTI systems, nonlinear MOR techniques are much less developed in terms of fidelity and efficiency. For example, for nonlinear systems, a reduced model is usually built based on a set of training input waveforms. Therefore, the reduced model is often only reliable for the same or similar inputs waveforms. Due to strong nonlinearity, a slightly change in the input waveform might result in unexpected discrepancy in the simulation results. Building a general and robust nonlinear reduced order model can be rather challenging. In the work of Kellems et al. (2010), nonlinear MOR was proposed to build reduced model for neurons with active dendrites. This is a nice work towards the right direction while many general nonlinear MOR challenges remain to be researched on.

To make reduced models more efficient and reliable, we isolate the active (nonlinear) and synaptic (time-varying) conductances. The remaining passive dendrite is an N-port network in the solid box. The N-port network is a multi-input multi-output (MIMO) LTI system. In this system, the inputs are port currents $i_1(t), i_2(t), \ldots, i_N(t)$ and the outputs are those port voltages $v_1(t), v_2(t), \ldots, v_N(t)$. Note that, although we are only interested on the voltage at somatic port $v_1(t)$, we still need to preserve the voltage responses at other ports $v_2(t), \ldots, v_N(t)$ to a certain degree. The reason is that the synaptic input currents are dependent on corresponding port voltages. For example, at Port 2, we have $i_2(t) = (E_2 - v_2(t))g_2(t)$. If $v_2(t)$ is not preserved, $i_2(t)$ in the next time step will be inaccurate, which will result in errors in the somatic voltage $v_1(t)$ eventually.

That is a big difference between passive dendrite reduction and quasi-active reduction. In the quasi-active case, there is no such dependence. As shown in Fig. 15, for quasi-active systems, we have $i_k(t) = (E_k - \bar{V}_k(t))g_k(t)$, where $\bar{V}_k(t)$ is the constant resting potential. As a result, given the inputs currents $i_1(t), i_2(t), \ldots, i_N(t)$, we only need to preserve the somatic voltage $v_1(t)$ and it is essentially a multi-input single output (MISO) system. Compared with MIMO systems, much more compact models can be obtained for MISO systems, which could be seen by comparing the results in Tables 1 and 4.

Model reduction for MIMO systems is challenging if the number of inputs and outputs are large (for the case of integration of a large number of synaptic inputs). This degradation is fundamental and does not depend on any particular reduction method. For example, for moment-matching methods, given the number of moments to be matched, the reduced order is proportional to the number of ports. Similarly, for balanced truncation methods, if the number of ports is larger, the decay of Hankel singular values will be slower and a larger reduced model is needed correspondingly. That is the reason why we propose Algorithm 2 instead of the classical Algorithm 1 to handle the case in Section 2.2.3. In Algorithm 2, only important port voltages are selected to be preserved. For example, given 100 ports, if Algorithm 1 is applied, it will result in a reduced model of order 500. However, Algorithms 2 only results in a reduced model of order 54, which gives excellent approximations for the output of interest, as shown in Fig. 12.

Second, we need to preserve passivity in the reduction. Under the context of linear system MOR, one important issue is the preservation of passivity in the reduced order model. A passivity-preserving method produces a passive reduced model if the full system is passive. This is a very desirable property as non-passive models can produce non-physical oscillation when interconnected with other circuit elements to form a larger network (Odabasioglu 1998). The proposed method can preserve passivity by exploiting the special structure of the state-space equations. The MOR approaches in Feldmann and Freund (1995) and Kellems et al. (2009) are not necessarily passive.

4.2 Impedance analysis of moment-matching

The reason why the reduced model by the proposed method can produce almost the same output waveforms as the full model is that the reduced model has the same impedance parameters as the full model in the frequency range of biological signals. To illustrate this point, we first show the impedances of reduced and full models as functions of frequency. Then we show

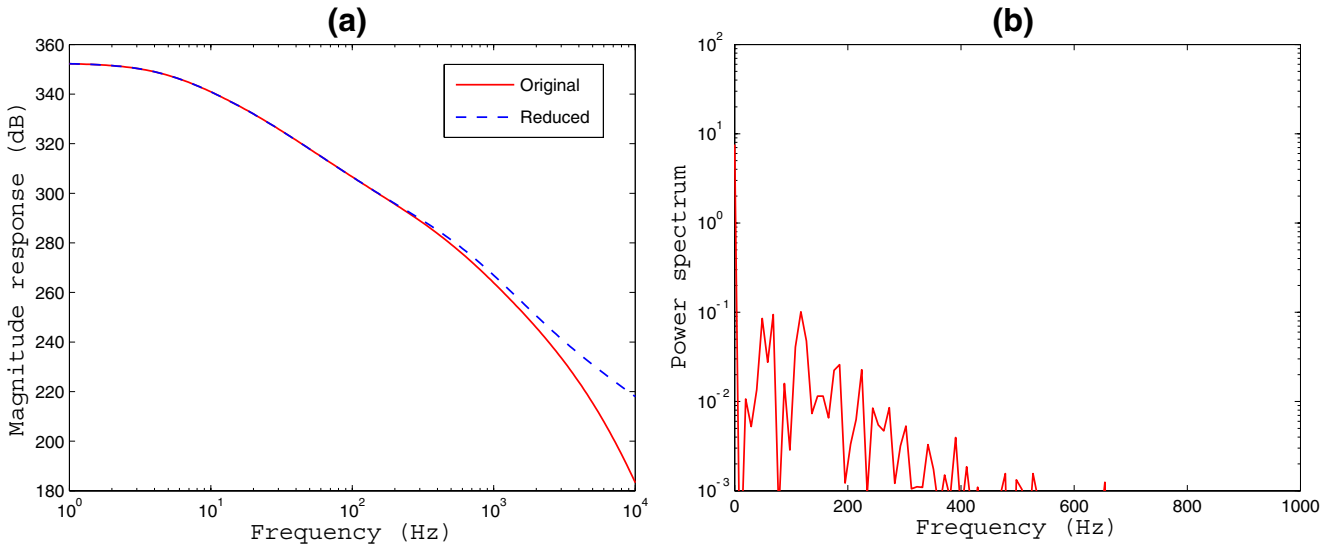

Fig. 18 The impedance of reduced and full models and the power spectrum density (PSD) of biological signal

the frequency components of the biological signals in neurons.

The impedance parameters of the passive dendrite are fully described by the Z-parameter matrix $\mathbf{Z}(s)$. The frequency response of the impedance can be obtain by evaluating $\mathbf{Z}(s)$ at $s = j\omega = j2\pi f$. Here, we use the somatic input impedance as an example, which is one of the entries of the matrix $\mathbf{Z}(\mathbf{s})$. As shown in Fig. 18(a), the impedances of the reduced and full models are indistinguishable at low frequencies due to moment matching around expansion point 0 Hz (Eq. (18)).

Now we consider the frequency components of biological signals in neurons. We use the action potential waveforms in Fig. 12(a) as an example. The power spectrum density (PSD) is given in Fig. 18(b). It is clear that the PSD is concentrated around 0 Hz and decays dramatically as the frequency increases. The PSD is about 1,000 times smaller as the frequency goes beyond 300 Hz and about 10,000 times smaller as the frequency goes beyond 500 Hz.

Comparing Fig. 18(a) and (b), we see the reason why the reduced model works very well: although the impedances of reduced and full models are quite different at high frequencies, the power of action potential signals is only concentrated at low frequencies and in this frequency range, the reduced model has the same impedance as the original model.

4.3 Comparison with existing techniques

Recently, a balanced truncation method (BT) (Moore 1981) has been proposed (Kellems et al. 2009) for model reduction of quasi-active systems as a benchmark. To demonstrate the accuracy of the proposed method, we compare it with BT. As BT is very expensive in terms of both reduction time and memory, a small example is used here. The quasi-active model of the small example is obtained by distributing Na and K channels with Hodgkin–Huxley kinetics uniformly on the realistic morphology of a 252 compartment dendrite of a Hormone cell (Roberts et al. 2010). The morphology is also available online in the database of Neuron simulator NeuronDB. After linearization, we get a quasi-active model of order 1008. We randomly choose 100 inputs and compare the accuracy of reduced models of both methods at different reduced orders.

As shown in Table 6, while the errors of reduced models are all very small, the proposed method is superior to balanced truncation in terms of accuracy. In addition, in this small example, while the reduction time for Algorithm 3 is 0.036 s, the reduction time for balanced truncation is 34.444 s, which is about 957 times more expensive. In fact, the cost of balanced truncation is $O(n^3)$, which is prohibitive for large examples.

Table 6 Comparison with balanced truncation

Reduced order	Max relative error		Mean relative error	
	BT (%)	Algorithm 3 (%)	BT (%)	Algorithm 3 (%)
1	0.585	0.202	0.484	0.190
2	0.328	0.072	0.265	0.068
3	0.160	0.038	0.126	0.021
4	0.051	0.016	0.048	0.013
5	0.047	0.012	0.011	0.002

Although BT is a very power technique to generate reduced models with wide band accuracy. The proposed method usually gives better results given the same reduced order. The reason is that the proposed method is based on moment-matching (Taylor expansion). Due to the fact that the spectrum of signal is concentrated at lower frequencies, Taylor expansion based method is preferred.

As BT is very expensive, a method based on H_2 approximation, IRKA (Gugercin et al. 2008), was proposed for large-scale systems to mitigate the computational cost. Although both Algorithm 3 and IRKA use Krylov subspaces as a numerical technique, they are based two different methodologies. The proposed method is based on moment-matching. The goal of IRKA is to minimize the L_2 norm of the error in the imaginary axis. Given the order of reduced model q, in order to solve the optimization problem, IRKA needs to interpolate the original system at the negative of q poles of original model. As the poles are unknown a priori, it makes an initial guess and uses Krylov subspace iteration to locate the poles until convergence. The use of Krylov subspace in IRKA is similar to the use of Krylov subspace iteration to find the eigenvalues of matrices.

Although Krylov subspaces are used in both algorithms, the proposed method is less expensive than IRKA. The cost of both methods are dominated by the number of matrix factorizations of the full system. The proposed method only takes 1 matrix factorization $\mathcal{G}^{-T}$. As for IRKA, if the reduced order is q, it takes $2q$ matrix factorizations in each iteration

$$(\mathcal{G}+\sigma_i\mathcal{C})^{-1}, \left(\mathcal{G}^T+\sigma_i\mathcal{C}^T\right)^{-1}, i=1,\ldots,q. \tag{40}$$

If it needs k times to converge, it will take $2qk$ matrix factorizations. Moreover, the performance of IRKA also depends on initial shift selection. Initialization strategies often require the spectrum information of the original system and thus add additional cost (Gugercin et al. 2008).

4.4 Application of the proposed techniques

The proposed method can be used to replace high order passive compartment models or quasi-active models with much smaller reduced models to speed up the analysis. The reduced models have very high fidelity in terms of input-output behavior and can be obtained efficiently. For quasi-active cases, some of the potential applications have been well demonstrated in Kellems et al. (2009).

Although the active properties of dendrites have been widely discovered, passive dendrite models are still valuable under many cases. For example, in Single and Borst (1998), the role of dendritic morphology in the process of visual motion detection was studied. In this work, a passive dendritic multi-compartmental model of a VS1 cell was reconstructed, which receives synaptic inputs from an array of Reichardt-type motion detectors (EMD). The model could reproduce the experimental results and theoretical predictions by showing how local modulations at each EMD are smoothed by integration in the dendritic tree to give a smoothed high-fidelity global output signal at the axon. Different from local dendritic signals, which indicate both direction and velocity of pattern motion, the output signal at the axon is purely directional selective. In this example, the proposed technique for passive reduction,

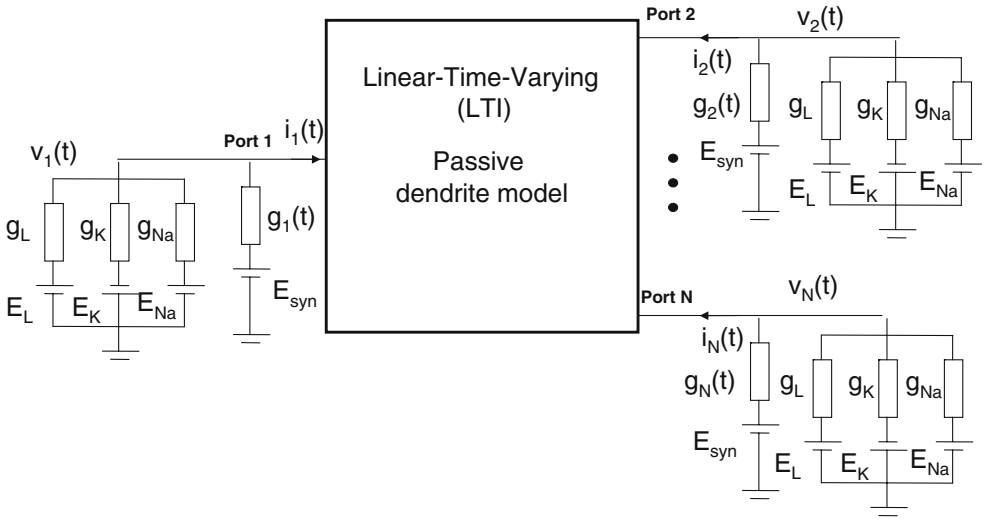

Fig. 19 Models with active dendrite conductances

Algorithm 2, can be used to replace the compartment model to speedup the analysis. This is especially necessary if the number of morphologies is large.

The proposed method could be useful to build large network models for the study of temporal codes. Temporal codes are based on precise timing of single spikes. They might be locked to external stimulus or intrinsically generated by the neural circuitry (Gerstner et al. 1997). In recent years, increasing experimental evidences have indicated that the firing rate concept based on temporal averaging may be too simplistic to describe brain activity (Stein et al. 2005). When precise spike timing is found to carry information, the neural code is often identified as a temporal code (Dayan and Abbott 2001). A number of studies have shown that the temporal resolution of the neural code is on a millisecond time scale, which means precise spike timing is a significant element in neural coding (Butts et al. 2007).

Note that, in this paper, reduced models have been obtained to match the exact somatic responses of the original model. However, the degree of reduction depends heavily on the fidelity required and the theoretical question asked. In the situation where the exact waveform is not needed, the reduced model can be made more compact and runs faster.

4.4.1 Extension to modeling active dendrites

The proposed method can be applied to reduce active dendrites. For example, as shown in Fig. 19, if Port $2-N$ have active conductances just as the Port 1 (somatic port), we can still isolate the nonlinear parts and reduce the remaining LTI system. There is no problem for us to include a small number of compartments with active conductances (just like soma). The limitation is that, if the number of compartments with active conductances is large, the amount of effective reduction is reduced.

Due to such limitations, for active cases, instead of high fidelity single neuron modeling, the proposed method is more suitable for generating highly simplified models for network simulation. In this case, given the passive dendritic compartment model, active conductances can be distributed in a small number of representative locations to reproduce the typical input-output behaviors of the neuron. As the number of such locations (Ports) could be small, a very compact reduced passive dendrite could be obtained. The reduced dendrite, connected the active conductances, can reproduce the responses of the original neuron exactly with significant speedup.

References

Antoulas, A. C. (2005). *Approximation of large-scale dynamical systems*. Philadelphia: Society of Industrial and Applied Mathematics (SIAM).

Bai, Z., & Su, Y. (2005). Dimension reduction of large-scale second-order dynamical systems va a second-order Arnoldi method. *SIAM Journal on Scientific Computing, 26*(5), 1692–1709.

Bower, J. M., & Beeman, D. (1998). *The book of GENESIS: Exploring realistic neural models with the GEneral NEural SImulation System*. New York: Springer.

Bush, P. C., & Sejnowski, T. J. (1993). Reduced compartmental models of neocortical pyramidal cells. *Journal of Neuroscience Methods, 46*, 159–166.

Butts, D. A., Weng, C., Jin, J., Yeh, C., Lesica, N. A., Alonso, J., et al. (2007). Temporal precision in the neural code and the timescales of natural vision. *Nature, 449*, 92–95.

Dayan, P., & Abbott, L. (2001), *Theoretical neuroscience: Computational and mathematical modeling of neural system*. Cambridge: MIT Press.

Djurfeldt, M., Lundqvist, M., Johansson, C., Rehn, M., Ekeberg, Ö., & Lansner, A. (2007). Brain-scale simulation of the neocortex on the IBM blue Gene/L supercomputer. *IBM Journal of Research and Development, 52*(1/2), 31–40.

Dyhrfjeld-Johnsen, J., Maier, J., Schubert, D., Staiger, J., Luhmann, H. J., Stephan, K. E., et al. (2005). CoCoDat: A database system for organizing and selecting quantitative data on single neurons and neuronal microcircuitry. *Journal of Neuroscience Methods, 141*, 291–308.

Feldmann, P., & Freund, R. W. (1995), Efficient linear circuit analysis by Padė approximation via the Lanczos process. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 14*, 639–649.

Gerstner, W., Kreiter, A. K., Markram, H., & Herz, A. V. M. (1997). Neural codes: Firing rates and beyond. *Proceedings of the National Academy of Sciences of the United States of America, 94*, 12740–12741.

Glover, K. (1984). All optimal Hankel norm approximations of linear multivariable systems and their L^{∞} error bounds. *International Journal of Control, 39*(6), 1145–1193.

Grimme, E. J., Sorensen, D. C., & Van Dooren, P. (2005). Model reduction of state space systems via an implicitly restarted Lanczos method. *Numerical Algorithms, 12*(1), 1–31.

Gugercin, S., Antoulas, S., & Beattie, C. (2008). H_2 model reduction for large-scale linear dynamical systems. *SIAM Journal on Matrix Analysis and Applications, 30*, 609–638.

Hodgkin, A., & Huxley, A. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *Journal of Physiology, 117*, 500–544.

Izhikevich, E. M., & Edelman, G. M. (2007). Large-scale model of mammalian thalamocortical systems. *Proceedings of the National Academy of Sciences of the United States of America, 105*, 3593–3598.

Jack, J. J. B., Noble, D., & Tsien, R. W. (1975). *Electric current flow in excitable cells.* Oxford: Calderon Press.

Kellems, A. R., Chaturantabut, S., Sorensen, D. C., & Cox, S. J. (2010). Morphologically accurate reduced order modeling of spiking neurons. *Journal of Computational Neuroscience, 28*, 477–494.

Kellems, A. R., Roos, D., Xiao, N., & Cox, S. J. (2009). Low-dimensional, morphologically accurate models of subthreshold membrane potential. *Journal of Computational Neuroscience, 27*, 161–176.

Koch, C. (1999). *Biophysics of computation*. Oxford: Oxford University Press.

Markram, H. (2006). The blue brain project. *Nature Reviews. Neuroscience, 7*, 153–160.

Moore, B. C. (1981). Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control, 26*, 17–31.

Odabasioglu, A. (1998). Prima: Passive reduced-order interconnect macromodeling algorithm. *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems, 17*, 645–654.

Pillage, L. T., & Rohrer, R. A. (1990). Asymptic waveform evaluation for timing analysis. *IEEE Transactions on Computer-aided Design of Integrated Circuits and Systems, 9*, 352–366.

Rall, W. (1959). Branching dendritic trees and motoneuron membrane resistivity. *Experimental Neurology, 1*, 491–527.

Rall, W. (1964). Theoretical significance of dendrite trees for neuronal input-output relations. In R. Reiss (Ed.), *Neuronal theory and modeling* (pp. 73–97). Stanford: Stanford University Press.

Rall, W. (1967). Distinguishing theoretical synaptic potentials computed for different soma-dendritic distribution of synaptic inputs. *Journal of Neurophysiology, 30*, 1138–1168.

Rapp, Y., Koch, C., & Segev, I. (1992). The impact of parallel fiber background activity on the cable properties of cerebellar purkinje cells. *Neural Computation, 4*, 518–532.

Roberts, C. B., Best, J. A., & Suter, K. J. (2010). Dendritic processing of excitatory synaptic input in hypothalamic gonadotropin releasing-hormone neurons. *Endocrinology, 147*, 1545–1555.

Saad, Y. (2003). *Iterative methods for sparse linear systems*. Philadelphia: Society of Industrial and Applied Mathematic (SIAM).

Salimbahrami, B., & Lohmann, B. (2002). Krylov subspace methods in linear model order reduction: Introduction and invariance properties. Scientific Report, Institute of Automation, University of Bremen.

Segev, I. (1992). Single neurone models: Oversimple, complex and reduced. *TINS 15*, 414–421.

Single, S., & Borst, A. (1998). Dendritic integration and its role in computing image velocity. *Science, 281*, 1848–1850.

Spruston, N. (2008). Pyramidal neurons: Dendritic structure and synaptic integration. *Nature Reviews. Neuroscience, 9*, 206–221.

Stein, R., Gossen, E., & Jones, K. (2005). Neuronal variability: noise or part of the signal? *Nature Reviews. Neuroscience, 6*, 389–397.

Stewart, G. W. (2001), *Matrix algorithms: Eigensystems*. Philadelphia: Society of Industrial and Applied Mathematic (SIAM).

Villemagne, C. D., & Skelton, R. E. (1987). Model reduction using a projection formulation. *International Journal of Control, 46*, 2141–2169.

Wilson, M. A., & Bower, J. M. (1989). The simulation of large scale neural networks. In C. Koch, & I. Segev (Eds.), *Methods in neuronal modeling* (pp. 291–333). Stanford: MIT Press.

Yan, B., Zhou, L., Tan, S., Chen, J., & McGaughy, B. (2008). DeMOR: Decentralized model order reduction of linear networks with massive ports. In *Proc. Design Automation Conf. (DAC)* (pp. 409–414).