31st Annual International Conference of the IEEE EMBS
Minneapolis, Minnesota, USA, September 2-6, 2009

Sparse Generalized Laguerre-Volterra Model of Neural Population Dynamics

Dong Song, *Member, IEEE*, Rosa H. M. Chan, *Student Member, IEEE*,
Vasilis Z. Marmarelis, *Fellow, IEEE*, Robert E. Hampson, Sam A. Deadwyler, and
Theodore W. Berger, *Senior Member, IEEE*

***Abstract*—To understand the function of a brain region, e.g., hippocampus, it is necessary to model its input-output property. Such a model can serve as the computational basis of the development of cortical prostheses restoring the transformation of population neural activities performed by the brain region. We formulate a sparse generalized Laguerre-Volterra model (SGLVM) for the multiple-input, multiple-output (MIMO) transformation of spike trains. A SGLVM consists of a set of feedforward Laguerre-Volterra kernels, a feedback Laguerre-Volterra kernel, and a *probit* link function. The sparse model representation involving only significant self and cross terms is achieved through statistical model selection and cross-validation methods. The SGLVM is applied successfully to the hippocampal CA3-CA1 population dynamics.**

I. INTRODUCTION

To understand the function of a brain region such as the hippocampus, it is necessary to model its input-output properties. The model should be able to predict accurately the output signal based on the on-going input signal and provide mechanistic insights into the systems nonlinear dynamics [1].

Volterra model is one of the best-known systems identification methods that has numerous successful applications on the modeling of physiological systems. It has many favorable properties in representing the systems nonlinear dynamics. However, Volterra model, at least in its original form, is not optimal for the point-process (spike) input-output characteristics of neural population dynamics. In this paper, we modify/extend the Volterra model based on the principal physiological mechanisms/processes of the spiking neuron and formulate a sparse generalized Laguerre-Volterra modeling (SGLVM) framework to better account for those characteristics. The SGLVM model is applied successfully to the hippocampal CA3-CA1 spike train transformations and may serve as the computational basis of the development of hippocampal prostheses.

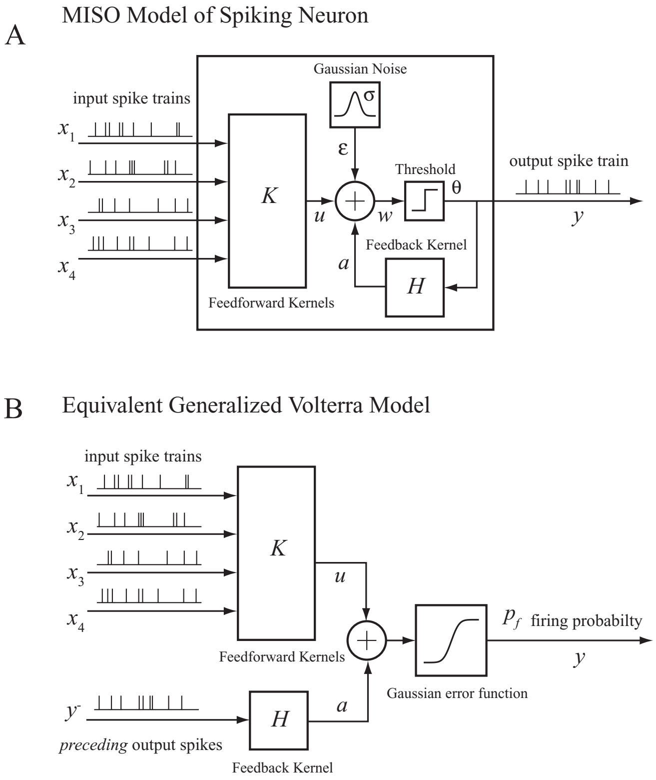

Fig. 1. Multiple-input single-output (MISO) spiking neuron model. A: structure of a MISO model. D: MISO model is equivalent to a generalized Volterra model with a *probit* link function..

II. METHODOLOGY

According to the Volterra modeling theory, a nonlinear time-invariant system with finite memory can be expressed in terms of the relation between progressively higher order temporal properties of the input events and the system output, and represented as kernels of a functional power series. Theoretically, a Volterra model (VM) can capture completely the input-output relation of such a system, given sufficiently high model order. In practice, due to the estimation difficulty associated with the requirement of forbiddingly long input-output dataset, VMs are often

Manuscript received April 12, 2007. This work was supported by NSF, DARPA(HAND), and NIH(NIBIB).

D. Song is with the Department of Biomedical Engineering, Center for Neural Engineering, University of Southern California, Los Angeles, CA 90089 USA (phone: 213-740-8063; fax: 213-740-0343; e-mail: dsong@usc.edu).

H. M. Chan, V. Z. Marmarelis, and T. W. Berger are with the Department of Biomedical Engineering, University of Southern California, Los Angeles, CA 90089 USA (e-mail: homchan@usc.edu, vzm@bmsr.usc.edu, berger@bmsr.usc.edu).

R. E. Hampson and S. A. Deadwyler are with the Department of Physiology & Pharmacology, Wake Forest University, School of Medicine, Winston-Salem, NC 27157 USA (e-mail: rhampson@wfubmc.edu, sdeadwyl@wfubmc.edu).

truncated to the low orders and the contributions of the higher-order terms are combined with the intrinsic system noise into a Gaussian random noise. In the case of a 2nd-order, N-input-single-output system:

$$\begin{aligned} y(t) = k_0 &+ \sum_{n=1}^{N} \sum_{\tau=0}^{M} k_1^{(n)}(\tau) x_n(t-\tau) \\ &+ \sum_{n=1}^{N} \sum_{\tau_1=0}^{M} \sum_{\tau_2=0}^{M} k_{2s}^{(n)}(\tau_1,\tau_2) x_n(t-\tau_1) x_n(t-\tau_2) \\ &+ \sum_{n_1=1}^{N} \sum_{n_2=1}^{n_1-1} \sum_{\tau_1=0}^{M} \sum_{\tau_2=0}^{M} k_{2x}^{(n_1,n_2)}(\tau_1,\tau_2) x_{n_1}(t-\tau_1) x_{n_2}(t-\tau_2) \\ &+ \varepsilon(t). \end{aligned} \tag{1}$$

$y(t)$ is the output; $x(t)$ is the input; k_0, k_1, k_{2s}, and k_{2x} are the zeroth-order, 1st-order, 2nd-order self, and 2nd-order cross kernels, respectively. M is the system memory. $\varepsilon(t)$ is a Gaussian random noise. The kernel coefficients then can be estimated using the least-squares method.

However, neurons generate spikes as output. Spikes are stochastic, discrete, fixed-amplitude events generated by a thresholding mechanism, and thus can be expressed as 1s without loss of generality (non-spiking events are expressed as 0s). They can be considered as realizations of a Bernoulli process with an underlying probability intensity function. The Gaussian noise assumption in Equation 1 is clearly inappropriate for such a probability distribution. To solve this problem, we modify the VM by adding a pre-threshold noise and a threshold function. In such a scheme, the output will be either 1s or 0s by definition, and the probability of generating 1s (crossing the threshold) is determined by the inputs and the noise term.

It is also well-known that the spiking activity of a neuron can be profoundly influenced by its previous spikes besides the input spikes it receives. Indeed, a neuron can generate drastically different spiking patterns with different types of spike-triggered dynamics given the same input pattern. To capture this characteristic, it is necessary to add an "auto-regressive" component that specifically models the output-dependent dynamics (the Volterra series in Equation 1 can be considered as a “moving-average” expression of the inputs). In fact, this is particularly important for a spiking neuron model since the latter generates 1-or-0 outputs that can not be deterministically derived from the input events. Not including such an auto-regressive component would not only mix the feedforward dynamics with the feedback dynamics and thus complicate the physiological interpretation, but also cause less accurate model by losing information about the exact timings of the previous output spikes. To address this issue, we further add a feedback term that transforms the *preceding* output spikes to a continuous hidden variable a that can be interpreted as spike-triggered after-potential. The resulting model is shown in Figure 1A and can be expressed as:

$$w = u(k,x) + a(h,y) + \varepsilon(\sigma) \tag{2}$$

$$y = \begin{cases} 0 & \text{when } w < \theta \\ 1 & \text{when } w \geq \theta \end{cases} \tag{3}$$

$$a(t) = \sum_{\tau=1}^{M} h(\tau) y(t-\tau) \tag{4}$$

The feedforward dynamics is represented by a hidden variable u that is expressed as a 2nd-order, N-input-single-output Volterra model as in Equation 1 (by replacing y with u). The feedback dynamics is denoted by another hidden variable a that is expressed as a 1st-order single-input-single-output Volterra model (Equation 4). ε is the pre-threshold noise that is assumed to be Gaussian distributed with a standard deviation of σ; w is the hidden variable (pre-threshold potential) that generates spikes when crossing the threshold θ.

One of the major challenges in Volterra modeling is the large number of open coefficients to be estimated. The total number of open parameters increases exponentially with input dimension and model order. The model involves 2-D (time and index of the input neurons) input and 2nd-order nonlinearity. The number of parameters easily becomes unwieldy even in a moderately large model. To solve this problem, we employ (1) Laguerre expansion of Volterra kernel (LEV), and (2) statistical model selection techniques.

Using LEV, both feedforward kernels k and feedback kernel h are expanded with orthonormal Laguerre basis functions b [2]. With input and output spike trains x and y convolved with b:

$$v_j^{(n)}(t) = \sum_{\tau=0}^{M} b_j(\tau) x_n(t-\tau) \ , \ v_j^{(h)}(t) = \sum_{\tau=1}^{M} b_j(\tau) y(t-\tau) \tag{5-6}$$

Equations 1 and 4 can be rewritten into:

$$\begin{aligned} u(t) = c_0 &+ \sum_{n=1}^{N} \sum_{j=1}^{L} c_1^{(n)}(j) v_j^{(n)}(t) \\ &+ \sum_{n=1}^{N} \sum_{j_1=1}^{L} \sum_{j_2=1}^{j_1} c_{2s}^{(n)}(j_1,j_2) v_{j_1}^{(n)}(t) v_{j_2}^{(n)}(t) \\ &+ \sum_{n_1=1}^{N} \sum_{n_2=1}^{n_1-1} \sum_{j_1=0}^{L} \sum_{j_2=0}^{L} c_{2x}^{(n_1,n_2)}(j_1,j_2) v_{j_1}^{(n)}(t) v_{j_2}^{(n)}(t) \end{aligned} \tag{7}$$

$$a(t) = \sum_{j=1}^{L} c_h(j) v_j^{(h)}(t) \tag{8}$$

c are the sought Laguerre expansion coefficients of k and h. Since the number of basis functions (L) can be made much smaller than the memory length (M), the number of open parameters is greatly reduced by the expansion.

With recorded input and output spike trains x and y, model parameters can be estimated using maximum-likelihood method. The negative log-likelihood function L can be expressed as:

$$L(y \mid x,k,h,\sigma,\theta) = -\sum_{t=0}^{T} \ln P(y \mid x,k,h,\sigma,\theta) \qquad (9)$$

where T is the data length, and P is the probability of generating the recorded output y:

$$P(y \mid x,k,h,\sigma,\theta) = \begin{cases} \text{Prob}\,(w \geq \theta \mid x,k,h,\sigma,\theta) & when \quad y=1 \\ \text{Prob}\,(w < \theta \mid x,k,h,\sigma,\theta) & when \quad y=0 \end{cases} \qquad (10)$$

Since ε is assumed to be Gaussian, the conditional firing probability intensity function P_f (the conditional probability of generating a spike at time t) can be calculated with the Gaussian error function (integral of Gaussian function) *erf*:

$$P_f(t) = 0.5 - 0.5erf\left(\frac{\theta - u(t) - a(t)}{\sqrt{2}\sigma}\right) \qquad (11)$$

where $$erf(s) = \frac{2}{\sqrt{\pi}}\int_0^s e^{-t^2}\,dt \qquad (12)$$

P at time t then can be calculated as:

$$P(t) = \begin{cases} p_f(t) & when \quad y=1 \\ 1 - p_f(t) & when \quad y=0 \end{cases} \qquad (13)$$

Model coefficients c then can be estimated by minimizing the negative log-likelihood function L:

$$\tilde{c} = \arg\min(L(c)) \qquad (14)$$

This model turns out to be mathematically equivalent to a Generalized Linear Model (GLM) with y as dependent variable, the convolutions of Laguerre basis functions with inputs x (v in Equation 7 and 8) as well as the products of the these convolutions (vv in Equation 7) as independent variables, and c as unknown parameters (Fig. 1B). The GLM link function is the *probit* function (inverse cumulative distribution function of the normal distribution) since the latter is defined as:

$$probit(y) = \sqrt{2}erf^{-1}(2y-1) \qquad (15)$$

Given this important equivalence, model coefficients c and their covariance matrices can be estimated using the iterative re-weighted least-squares method, the standard algorithm for fitting GLMs. For the same reason, this model can be termed as Generalized Laguerre-Volterra Model (GLVM). Since u, a and n are dimensionless variables, without loss of generality, both θ and σ can be set to unity value; only c are estimated.

Theoretically, the aforedescribed method can be used to estimate arbitrary multiple-input models. However, in practice, model complexity often needs to be further reduced by selecting an optimal subset of model coefficients. This procedure, termed model selection, is particularly necessary and desirable in modeling the population neural dynamics for the following reasons: first, neurons are often sparsely connected. In a brain region, an output neuron is seldom affected by all the input neurons. The full Volterra kernel model as describe in Equation 1 is not the most efficient or interpretable way of representing such system. More importantly, the number of coefficients to be estimated in a full Volterra kernel model grows rapidly with the number of inputs and the model order. Estimation of such model, especially the higher order ones, can easily become unwieldy even with the Laguerre expansion. Furthermore, a model with too many open coefficients tends to fit the noise instead of the signal in the training data. An overfitting model would result in poor generalization of the training data and bad predictions of the novel data. Consequently, interpretation of such model becomes problematic. To solve this problem, the following statistical model selection method is applied to the configuration and estimation of the GLVMs.

Before model selection, the input-output dataset is partitioned into two subsets. One subset (training set) is used for model estimation. The other subset (testing set) is retained for validation of the results from the training set. Results from the two subsets are called in-sample and out-of-sample results, respectively. The model starts from zeroth order. A zeroth order model only contains c_0, which is equivalent to the standard deviation of the pre-threshold Gaussian noise. It essentially model the system output as a homogeneous Poisson process (constant firing probability intensity). The minimal negative likelihood (L) of zeroth order model provides a starting point for the model selection. In the second step, feedback terms (as described by Equation 8) is added to the model. Output spike train is predicted by the preceding output spikes without considering any input. If L decreases in both of the in-sample and out-of-sample results, the feedback term is then added into the model. In the third step, inputs are selected using a forward step-wise selection procedure. Self-terms involving first-order and second-order kernels are constructed for all inputs. With the zeroth order term and feedback term (if selected in the previous step) included in the model, the values of L with and without each input are calculated. The input that decreases L the most is then added into the model. With the newly selected input included into the model, selection is then performed on the remaining inputs. Repeating this procedure, inputs are sequentially added into the model. The selection is stopped when the out-of-sample L starts to increase (in-sample L always decreases with more terms included in the model), indicating the occurrence of overfitting. In the fourth step, cross-terms involving cross-kernel are selected. Cross-terms are first constructed for every unique pairs of the selected inputs and then selected following the aforedescribed forward step-wise and cross-validation procedures. The resulting model represents the system with sparse model coefficients corresponding only to the significant model terms and thus can be termed as a sparse GLVM (SGLVM).

III. RESULTS

SGLVM is applied on the hippocampal CA3-CA1 population dynamics in rats performing a memory-dependent behavior task, i.e., delayed-nonmatch-to-sample task [3]. CA3 and CA1 spike trains are simultaneously recorded during the task and constitute the MIMO datasets for the model.

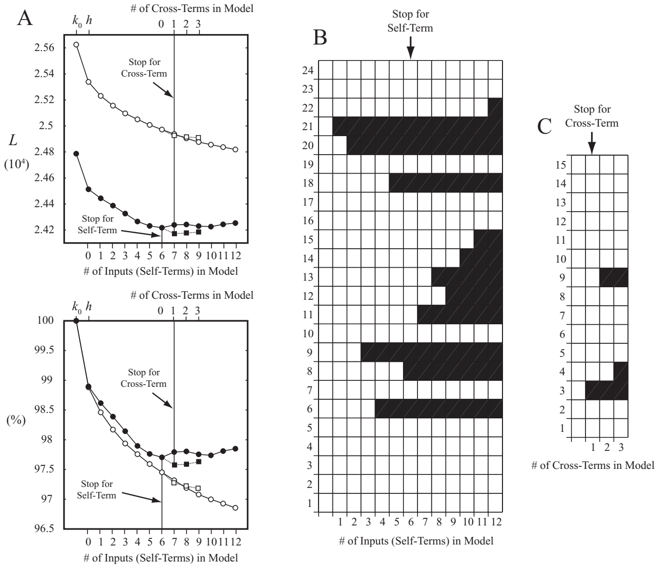

Fig. 2. Selecting significant inputs and cross-terms of a MISO model. A: The model starts from zeroth order (first data points). Feedback kernels is then selected and included in the model (second data points). Six inputs (first and second order self-terms) and one cross-term are selected based on cross-validation. Open circles: in-sample results; Closed circles: out-of-sample results; Top: absolute negative log-likelihood (*L*); Bottom: normalized *L*. B: Left, selection path of the inputs; Right: selecting path of the cross-terms. The cross-terms (# = 15) are constructed with the selected inputs (# = 6) only.

MISO SGLVMs are obtained using the aforedescribed estimation and selection method. Figure 2 illustrates the selection of inputs and cross-terms of a SGLVM. The model starts from zeroth order. The feedback term is then added into the model since it decreases L in both training and testing datasets (Fig. 2A). With the zeroth order and feedback terms, inputs are added into the model in a forward step-wise fashion (Fig. 2B). The in-sample L decreases monotonically as more terms added into the model. However, the out-of-sample L starts to increase from the 7th input. Six inputs are then selected based on this cross-validation results (Fig. 2A). With the 6 selected inputs, 15 cross-terms in total are then constructed and added into the model (Fig. 2C). Again, in-sample L decreases monotonically; out-of-sample L starts to increase from the 2nd cross-term – so only one cross-term is selected by cross-validation (Fig. 2A). It is evident that the model selection procedure greatly reduces the model complexity and thus allows reliable estimation of the model.

The estimated model is validated using out-of-sample Kolmogorov-Smirnov (KS) test based on the time-rescaling theorem [4]. The summation of synaptic potential u and after-potential a is calculated using the input/output spike trains of the testing set, and the kernels estimated from the training set. Firing probability intensity P_f is then calculated using the Gaussian error function. The KS plot shows that all data points are within the 95% confidence bound.

IV. DISCUSSION

We have formulated a SGLVM framework for the population neural dynamics and applied it successfully to the modeling of hippocampal CA3-CA1 spike train transformation. The model inherits the capability of modeling nonlinear dynamic systems from the ordinary Volterra model while having several critical modifications/improvements made specific for the modeling of neural population activity.

The model described in this paper can be used to build hippocampal prostheses. For example, to replace a CA1 cell field, the prosthetic device has to reinstate the output signal (e.g., CA1 spikes) based on the signals recorded in an upstream region (e.g., CA3). A prosthesis including our CA3-CA1 model would be used in the following manner. First, stimulating electrodes would be placed in the molecular layer of the subiculum. CA3 neurons would provide input to the model. Thus, spike output of the model would provide the output of simulated CA1 to the subiculum, the output target of CA1. This by-passes the damaged CA1 area, and restores CA3-to-subiculum nonlinear dynamics, and thus, the “near-normal” nonlinear dynamic output of the hippocampus.

REFERENCES

[1] D. Song, R. H. M. Chan, V. Z. Marmarelis, R. E. Hampson, S. A. Deadwyler, and T. W. Berger "Nonlinear dynamic modeling of spike train transformations for hippocampal-cortical prostheses," *IEEE Transactions on Biomedical Engineering*, vol. 54, 2007.

[2] V. Z. Marmarelis, “Identification of nonlinear biological systems using Laguerre expansions of kernels,” *Annals of Biomedical Engineering*, vol. 21, pp. 573-589, 1993.

[3] R. E. Hampson, J. D. Simeral, and S. A. Deadwyler, "Distribution of spatial and nonspatial information in dorsal hippocampus," *Nature*, vol. 402, pp. 610-4, 1999.

[4] E. N. Brown, R. Barbieri, V. Ventura, R. E. Kass, and L. M. Frank, "The time-rescaling theorem and its application to neural spike train data analysis," *Neural Computation*, vol. 14, pp. 325-46, 2002.