Annals of Biomedical Engineering, Vol. 30, pp. 272–281, 2002
Printed in the USA. All rights reserved.

0090-6964/2002/30(2)/272/10/$15.00

Modeling of Nonlinear Physiological Systems with Fast and Slow Dynamics. I. Methodology

G. D. MITSIS and V. Z. MARMARELIS

Department of Biomedical Engineering, University of Southern California, University Park, Los Angeles, CA

(Received 30 April 2001; accepted 2 November 2001)

Abstract—Effective modeling of nonlinear dynamic systems can be achieved by employing Laguerre expansions and feedforward artificial neural networks in the form of the Laguerre–Volterra network (LVN). This paper presents a different formulation of the LVN that can be employed to model nonlinear systems displaying complex dynamics effectively. This is achieved by using two different filter banks, instead of one as in the original definition of the LVN, in the input stage and selecting their structural parameters in an appropriate way. Results from simulated systems show that this method can yield accurate nonlinear models of Volterra systems, even when considerable noise is present, separating at the same time the fast from the slow components of these systems effectively. © *2002 Biomedical Engineering Society.*
[DOI: 10.1114/1.1458591]

Keywords—Nonlinear modeling, Volterra models, Artificial neural networks, Polynomial activation functions, Discrete-time Laguerre functions.

INTRODUCTION

In the mathematical modeling of physiological systems, a practical problem arises when the dynamics of the system cover vastly different time scales (slow and fast), because the fast time scale requires a high sampling frequency and the slow time scale necessitates long experimentation times. The combination of these two requirements leads to very large amounts of stimulus-response data and presents a serious challenge to the efficacy of the estimation methods used for system identification and modeling. This problem is further aggravated by the presence of nonlinearities, since the latter impose additional burdens of accuracy and sensitivity on the estimation methods.

The subject paper presents a novel methodology by which this problem can be effectively resolved in the nonlinear context. The methodology employs a variant of the recently introduced Laguerre–Volterra networks for modeling nonlinear systems,[1] whereby two Laguerre filter banks are used with distinct trainable parameters ''alpha'' (corresponding to the slow and fast dynamics of the system). It is shown that this method addresses the aforementioned problem in a practical and effective manner.

Previous modeling studies of nonlinear Volterra systems demonstrated the practical advantages of using Laguerre expansions of the Volterra kernels in order to achieve model compactness and estimation accuracy.[5] The resulting Laguerre expansion technique (LET) can be combined with feedforward artificial neural networks utilizing polynomial activation functions in the form of the Laguerre–Volterra network (LVN). The latter receives as its input vector the outputs of a Laguerre filter bank fed by the input signal of the system.[1] The performance of the LET and LVN modeling methods has been shown to be excellent as long as the Laguerre parameter α is properly selected, since the latter determines the required number of Laguerre functions for the kernel expansion, i.e., determines the modeling parsimony and efficiency. An adaptive estimation method has been recently introduced for selecting the Laguerre parameter α $(0<\alpha<1)$ using the actual stimulus-response data (submitted for publication).

It is critical to note that the Laguerre parameter α defines the time scale for which the Laguerre expansion of the system kernels is most efficient in terms of convergence (i.e., yields satisfactory approximations with the minimum number of Laguerre basis functions). Thus, efficient use of the LET or LVN method requires a small α (close to 0) for systems with fast dynamics or a large α (close to 1) for systems with slow dynamics.

In the case of systems with both fast and slow dynamics, the choice of a single parameter α cannot lead to an efficient representation of the system and may not produce satisfactory results. Therefore, the use of two Laguerre filter banks characterized by two distinct Laguerre parameters holds the promise that the fast and slow components of a system can be modeled separately and effectively. The proposed model representation also captures the possible nonlinear interactions between slow and fast dynamics in the form of cross-terms. The proposed methodology is presented in the next sec-

Address correspondence to Professor V. Z. Marmarelis, OHE 500, USC, Los Angeles, CA 90089-1451. Electronic mail: vzm@bmsrs.usc.edu

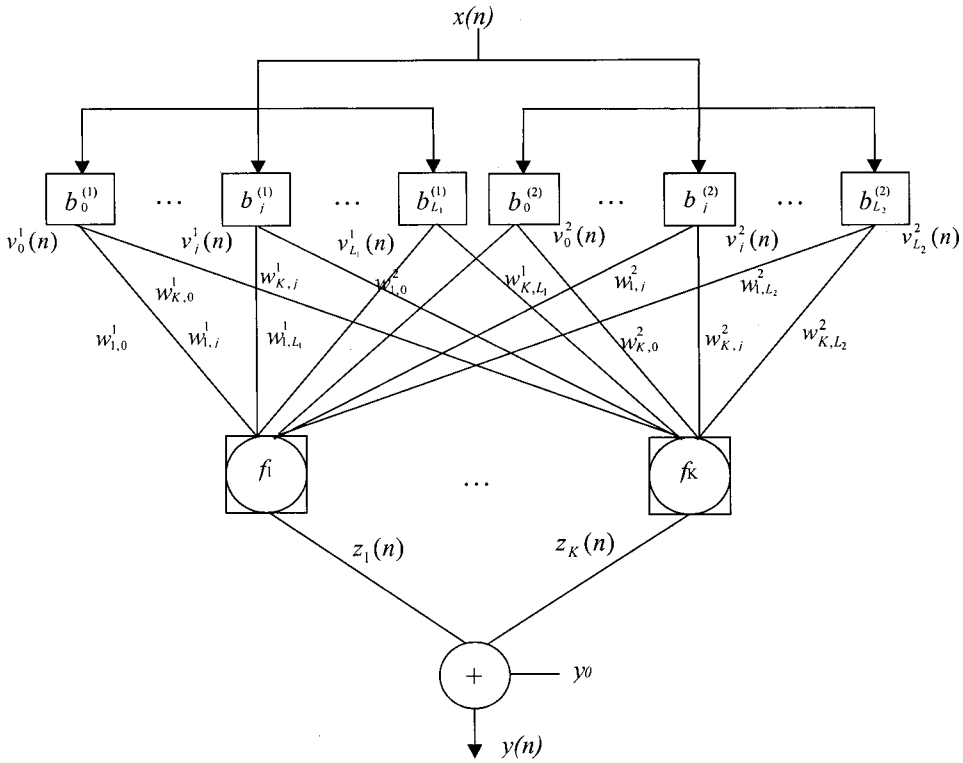

FIGURE 1. The LVN with two Laguerre filter banks $\{b_j^{(1)}\}$ and $\{b_j^{(2)}\}$ that preprocess the input $x(n)$. The hidden units in the second layer have polynomial activation functions $\{f_k\}$ and receive input from all Laguerre filters. The output $y(n)$ is formed by summation of the outputs of the hidden units $\{z_k\}$ and the output offset y_0.

tion. Illustrative examples with synthetic data are given in the corresponding section. An application of this method to actual experimental data of cerebral hemodynamics in humans is presented in a companion paper (submitted for publication as Part II).

METHODOLOGY

The proposed architecture of the Laguerre–Volterra network with two Laguerre filter banks (LVN-2) that preprocess the input is shown in Fig. 1. The two filter banks are characterized by different Laguerre parameters (α_1, α_2) and may contain different numbers of Laguerre functions (filters). By assigning a small value to α_1 for the first filter bank and a large value to α_2 for the second one, we can simultaneously model the fast and the slow components of a system separately.

The asymptotically exponential structure of the Laguerre functions makes them a good choice for modeling physiological systems, since the latter often exhibit asymptotically exponential structure in their Volterra kernels. However, one cannot rule out the possibility of system kernels that do not decay smoothly—a situation that will require either a large number of Laguerre functions or an alternate (more suitable) filter bank. The Laguerre parameter α defines the relaxation rate of the Laguerre functions and determines the convergence of the Laguerre expansion for a given kernel function. Larger α values result in longer spread of significant values (slow dynamics).

The choice of the Laguerre parameters (α_1, α_2) for the two filter banks is very important in order to achieve an efficient model representation of the system under examination. However, this choice has been made heretofore by trial-and-error procedures.[1,2,10] We recently introduced a computationally efficient method, whereby the Laguerre parameter is treated as a trainable parameter of the Laguerre–Volterra network (submitted for publication). This automates the procedure for the determination of suitable Laguerre parameters, guided by the actual experimental data.

The discrete-time Laguerre functions employed in the filter-banks are defined for the jth order as[8]

$$b_j^{(i)}(m) = \alpha_i^{(m-j)/2}(1-\alpha_i)^{1/2}\sum_{k=0}^{j}(-1)^k\binom{m}{k}\binom{j}{k}\alpha_i^{j-k}(1-\alpha_i)^k; \quad i=1,2, \tag{1}$$

where α_1, α_2 are the Laguerre parameters of the two filter banks (both lying between 0 and 1). The corresponding Laguerre filter output is given by the discrete convolution of $b_j^{(i)}(m)$ with the input signal $x(n)$:

$$\nu_j^{(i)}(n) = \sum_{m=0}^{\infty} b_j^{(i)}(m) x(n-m); \quad i = 1,2. \tag{2}$$

The hidden units in the second layer have polynomial (instead of the commonly used sigmoidal) activation functions in order to make the network functionally equivalent to a Volterra model and reduce the required number of hidden units.[7] The input of the kth hidden unit is given by

$$u_k(n) = \sum_{i=1}^{2} \sum_{j=0}^{L_i} w_{k,j}^{(i)} \nu_j^{(i)}(n); \quad k = 1,2,...,K \tag{3}$$

and its corresponding output is

$$z_k(n) = \sum_{m=1}^{Q} c_{m,k} u_k^m(n); \quad k = 1,2,...,K. \tag{4}$$

The output of the network is given by a nonweighted summation of all the hidden-unit outputs (since the polynomial coefficients are trainable) and a trainable offset y_0:

$$y(n) = \sum_{k=1}^{K} z_k(n) + y_0. \tag{5}$$

The training of all the network parameters is performed using the backpropagation algorithm in an iterative fashion.[3] The amount by which a specific parameter value is changed in each iteration is proportional to the value of the partial derivative of the cost function with respect to the specific parameter at that iteration. Defining the cost function $J(n)$ as the squared error $\varepsilon(n)$ between the desired output $d(n)$ and the network output $y(n)$ at each time instance n:

$$J(n) = \tfrac{1}{2}[d(n) - y(n)]^2 \tag{6}$$

the iterative relations become

$$w_{k,j_i}^{(i),(r+1)} = w_{k,j_i}^{(i),(r)} - \gamma_w \left(\frac{\partial J}{\partial w_{k,j_i}^{(i)}} \right)_r = w_{k,j_i}^{(i),(r)} + \gamma_w [\varepsilon(n) f_k'(u_k(n)) \nu_{j_i}^{(i)}(n)]_r, \tag{7}$$

$$c_{m,k}^{(r+1)} = c_{m,k}^{(r)} - \gamma_c \left(\frac{\partial J}{\partial c_{m,k}} \right)_r = c_{m,k}^{(r)} + \gamma_C [\varepsilon(n) u_k^m(n)]_r, \tag{8}$$

$$y_0^{(r+1)} = y_0^{(r)} - \gamma_y \left(\frac{\partial J}{\partial y_0} \right)_r = y_0^{(r)} + \gamma_y \varepsilon^{(r)}(n), \tag{9}$$

$$\alpha_i^{(r+1)} = \alpha_i^{(r)} - \gamma_i \left(\frac{\partial J}{\partial \alpha_i} \right)_r = \alpha_i^{(r)} + \gamma_i \varepsilon^{(r)}(n) \sum_{k=1}^{K} \sum_{m=1}^{Q} \sum_{j=0}^{L_i} m \times \left[c_{m,k} w_{k,j}^{(i)} u_k^{m-1}(n) \frac{\partial \nu_j^{(i)}(n)}{\partial \alpha_i} \right]_r, \tag{10}$$

where r denotes the iteration index and $\gamma_w, \gamma_c, \gamma_y, \gamma_i$ are fixed positive learning constants. Note that $\varepsilon^{(r)}(n)$ is the output error at the rth iteration and $f_k'^{(r)}$ is the derivative of the polynomial activation function of the kth hidden unit at the rth iteration

$$f_k'^{(r)}[u_k^{(r)}(n)] = \sum_{m=1}^{Q} m c_{m,k} [u_k^{(r)}(n)]^{m-1}. \tag{11}$$

The total number of unknown parameters in the LVN-2 model is equal to $(L_1 + L_2 + 2 + Q)K + 3$. It is important to note here that this number is linear with respect to the order Q of the system, which makes this approach more practical for higher-order systems compared to other techniques (e.g., cross-correlation), the complexity of which depends on the system order Q exponentially.

The difficulty in training the parameter α based on the above Eq. (10) is evident, since the use of the backpropagation algorithm requires computation of the partial derivatives of each $\nu_j^{(i)}(n)$ with respect to α_i. As can be seen from Eq. (1), $b_j^{(i)}(m)$ depends in a complex way on α_i, which makes this computation based on Eq. (2) rather excessive for all practical purposes.

However, this task becomes greatly simplified if we employ the recursive relations for the computation of the Laguerre filter-bank outputs. Specifically, the output $\nu_j^{(i)}(n)$ of the jth discrete-time Laguerre filter of the ith filter-bank can be expressed as follows:

$$\nu_j^{(i)}(n) = \beta_i [\nu_j^{(i)}(n-1) + \nu_{j-1}^{(i)}(n)] - \nu_{j-1}^{(i)}(n-1), \tag{12}$$

where $\beta_i = \alpha_i^{1/2}$.

The initial expression (for $j = 0$) is slightly different:

$$\nu_0^{(i)}(n)=\beta_i\nu_0^{(i)}(n-1)+(1-\beta_i^2)^{1/2}x(n). \quad (13)$$

These recursive relations are obtained from the Z transforms of the discrete-time Laguerre functions.[8]

We can see that it is easier to train β_i rather than α_i based on the iterative relations (12) and (13). This can be done by differentiating Eq. (12) with respect to β_i:

$$\frac{\partial\nu_{j_i}^{(i)}(n)}{\partial\beta_i}=\nu_{j_i}^{(i)}(n-1)+\nu_{j_i-1}^{(i)}(n); \quad j_i=1,...,L_i \quad (14)$$

with the following initial condition ($j_i=0$):

$$\frac{\partial\nu_0^{(i)}(n)}{\partial\beta_i}=\nu_0^{(i)}(n-1)-\frac{\beta_i}{\sqrt{1-\beta_i}}x(n). \quad (15)$$

Then, the iterative expression for the training of β_i becomes (for $j_i>0$):

$$\beta_i^{(r+1)}=\beta_i^{(r)}+\gamma_i\varepsilon^{(r)}(n)\sum_{k=1}^{K}\sum_{m=1}^{Q}m\left\{c_{m,k}u_k^{m-1}(n)\times\sum_{j=0}^{L_i}w_{k,j}^{(i)}[\nu_{j_i}^{(i)}(n-1)+\nu_{j_i-1}^{(i)}(n)]\right\}_r. \quad (16)$$

In order to assist the reader in making the formal connection between the LVN-2 and the Volterra models, let us now consider the Volterra model for a Qth order nonlinear time-invariant discrete-time system

$$y(n)=\sum_{n=0}^{Q}\left[\sum_{m_1}\dots\sum_{m_n}k_n(m_1,..,m_n)x(n-m_1)\dots x(n-m_n)\right], \quad (17)$$

where $k_n(m_1,\dots,m_n)$ is the nth order Volterra kernel of the system. It can be shown that this representation is equivalent with the LVN-2 representation having activation functions of Qth degree in the hidden units. The Volterra kernels corresponding to the LVN-2 can be expressed in terms of the network parameters in the following way:

$$k_0=y_0, \quad (18)$$

$$k_1(m_1)=\sum_{i=1}^{2}\sum_{k=1}^{K}c_{1,k}\sum_{j=0}^{L_i}w_{k,j}^{(i)}b_j^{(i)}(m_1), \quad (19)$$

$$k_2(m_1,m_2)=\sum_{i_1=1}^{2}\sum_{i_2=1}^{2}\sum_{k=1}^{K}c_{2,k}\sum_{j_1=0}^{L_{i_1}}\times\sum_{j_2=0}^{L_{i_2}}w_{k,j_1}^{(i_1)}b_{j_1}^{(i_1)}(m_1)w_{k,j_2}^{(i_2)}b_{j_2}^{(i_2)}(m_2), \quad (20)$$

$$k_n(m_1,\dots,m_n)=\sum_{i_1=1}^{2}\dots\sum_{i_n=1}^{2}\times\sum_{k=1}^{K}c_{n,k}\sum_{j_1=0}^{L_{i_1}}\dots\times\sum_{j_n=0}^{L_{i_n}}w_{k,j_1}^{(i_1)}\dots w_{k,j_n}^{(i_n)}b_{j_1}^{(i_1)}(m_1)\dots b_{j_n}^{(i_n)}(m_n). \quad (21)$$

From Eq. (19) we can see that the first-order fast ($i=1$) and slow ($i=2$) components of a nonlinear system can be expressed in terms of the network parameters separately:

$$k_{1,f}(m_1)=\sum_{k=1}^{K}c_{1,k}\sum_{j=0}^{L_1}w_{k,j}^{(1)}b_j^{(1)}(m_1), \quad (22)$$

$$k_{1,S}(m_1)=\sum_{k=1}^{K}c_{1,k}\sum_{j=0}^{L_2}w_{k,j}^{(2)}b_j^{(2)}(m_1). \quad (23)$$

For the higher order kernels, in addition to separate slow and fast components, there exist cross-terms that represent nonlinear interactions between slow and fast dynamics. For example, in the case of the second-order kernel the following terms correspond to the fast and slow components separately:

$$k_{2,ff}(m_1,m_2)=\sum_{k=1}^{K}c_{2,k}\sum_{j_1=0}^{L_1}\sum_{j_2=0}^{L_1}w_{k,j_1}^{(1)}b_{j_1}^{(1)}(m_1)w_{k,j_2}^{(1)}b_{j_2}^{(1)}(m_2), \quad (24)$$

$$k_{2,SS}(m_1,m_2)=\sum_{k=1}^{K}c_{2,k}\sum_{j_1=0}^{L_2}\sum_{j_2=0}^{L_2}w_{k,j_1}^{(2)}b_{j_1}^{(2)}(m_1)w_{k,j_2}^{(2)}b_{j_2}^{(2)}(m_2), \quad (25)$$

and the following cross-term represents second-order interactions between fast and slow dynamics:

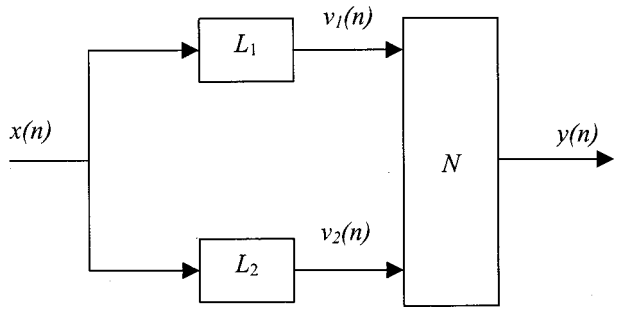

FIGURE 2. First simulated nonlinear system.

$$k_{2,fS}(m_1,m_2)=\frac{1}{2}\sum_{k=1}^{K} c_{2,k} \times \sum_{j_1=0}^{L_1}\sum_{j_2=0}^{L_2}[w_{k,j_1}^{(1)}b_{j_1}^{(1)}(m_1)w_{k,j_2}^{(2)}b_{j_2}^{(2)}(m_2) + w_{k,j_1}^{(1)}b_{j_1}^{(1)}(m_2)w_{k,j_2}^{(2)}b_{j_2}^{(2)}(m_1)]. \quad (26)$$

The second-order kernel is the summation of the earlier three components. Similar decompositions apply for the higher order kernels as well. It is evident that there is a wealth of information in these kernel components that cannot be retrieved with any other existing method.

A critical practical issue for the successful application of the LVN-2 is the proper selection of its structural parameters, i.e., the size of the Laguerre filter banks [number of Laguerre functions (LFs) L_1 and L_2], the number K of hidden units, and the degree Q of the polynomial activation functions. This selection can be performed by successive trials in ascending order (i.e., moving from lower to higher numbers) using a minimum description length (MDL) criterion for the mean-square error (MSE) of the output prediction achieved by the model.[9] Specifically, we commence the LVN-2 training with structural parameter values: $L_1=L_2=1$, $K=1$, $Q=1$ and compute the output MSE value for the trained LVN-2 model [i.e., the sum of all $J(n)$ in the output data record]. Then the structural parameters are incremented sequentially (starting with L_1 and L_2 concurrently, and continuing with K and Q) until the MDL criterion is met.

ILLUSTRATIVE EXAMPLES

The performance of the proposed model/network was evaluated initially with three simulated nonlinear systems. The first system is shown in Fig. 2, where L_1 and L_2 are linear filters characterized by the following impulse response functions:

$$l_1(m)=b_0^{(1)}(m)+2b_1^{(1)}(m)+b_2^{(1)}(m), \quad (27)$$

$$l_2(m)=b_0^{(2)}(m)-b_1^{(2)}(m)+2b_2^{(2)}(m), \quad (28)$$

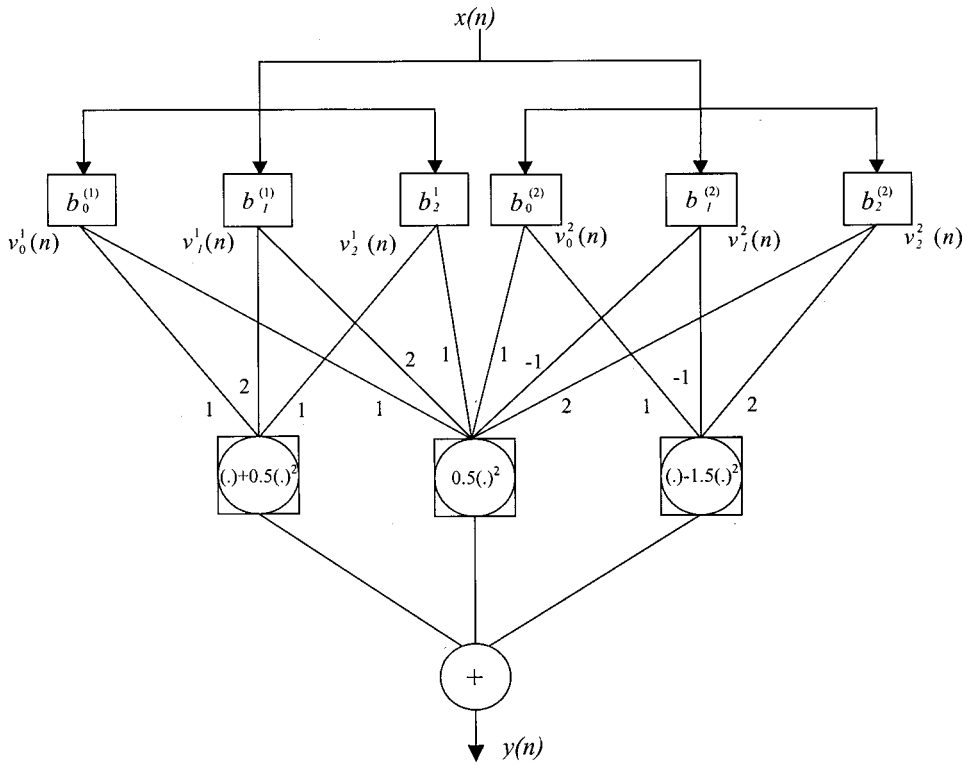

FIGURE 3. Equivalent LVN for the first simulated system.

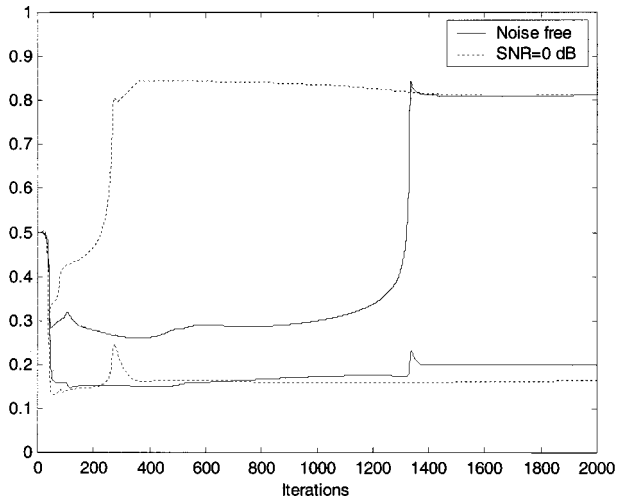

FIGURE 4. The learning curves for the two Laguerre parameters for the first simulated system (exact values are $\alpha_1 = 0.2$, $\alpha_2 = 0.8$).

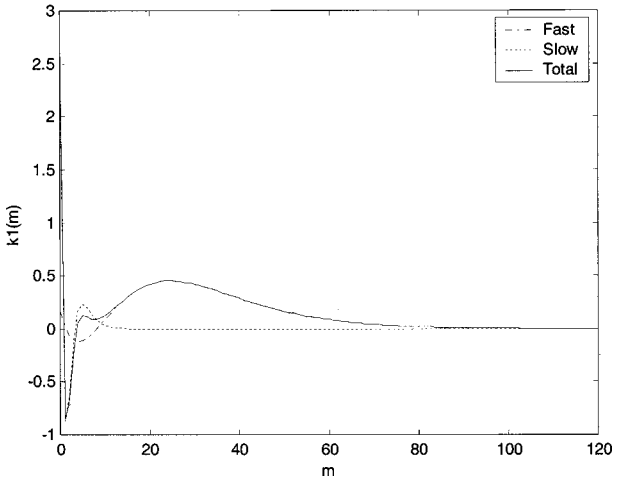

FIGURE 5. Estimated first-order kernel for the first simulated system and its slow/fast components.

composed of linear combinations of the first three discrete-time Laguerre functions, $b_0^{(i)}(m), b_1^{(i)}(m), b_2^{(i)}(m)$ $(i=1,2)$ with distinct Laguerre parameters: $\alpha_1 = 0.2$ and $\alpha_2 = 0.8$. The static polynomial nonlinearity N is given by

$$y(n) = \nu_1(n) + \nu_2(n) + \nu_1^2(n) - \nu_2^2(n) + \nu_1(n)\nu_2(n), \tag{29}$$

where ν_1 and ν_2 are the outputs of L_1 and L_2, respectively.

The first and second-order Volterra kernels of the system are

$$k_1(m) = l_1(m) + l_2(m), \tag{30}$$

$$k_2(m_1, m_2) = l_1(m_1)l_1(m_2) - l_2(m_1)l_2(m_2) + \tfrac{1}{2}[l_1(m_1)l_2(m_2) + l_1(m_2)l_2(m_1)]. \tag{31}$$

It is evident from these expressions that the two kernels contain fast and slow components that correspond to the two distinct Laguerre parameters.

TABLE 1. LVN and LVN-2 model performance; noise-free output.

			Prediction	Kernel NMSEs	
	α_1	α_2	NMSE (%)	$k_1(m)$ (%)	$k_2(m_1, m_2)$ (%)
LVN-2	0.200	0.809	0.43	0.18	0.13
LVN	0.528	⋯	14.28	10	14.67

This system is simulated for a Gaussian white noise (GWN) stimulus of unit variance and a length of 1024 points. Following the ascending-order search procedure described earlier, we determine that three LFs in each filter bank $(L_1 = L_2 = 2)$ and three hidden units $(K = 3)$ with distinct second-degree polynomial activation functions $(Q = 2)$ are adequate to model the system. This was theoretically anticipated because this system has the equivalent network model shown in Fig. 3. At least three hidden units are needed to express the output in terms of the linear filter outputs, due to the presence of all possible first and second-order terms and cross-terms in Eq. (29). Note that the selection of the parameter values in the equivalent LVN-2 is not unique and that only one of the possible selections is shown in Fig. 3, corresponding

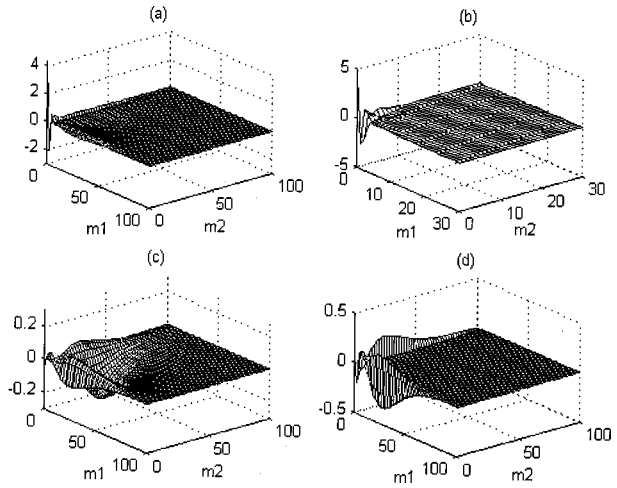

FIGURE 6. Estimated second-order kernel for the first simulated system (a), and its three components: (b) fast component, (c) slow component, and (d) cross component.

TABLE 2. LVN-2 and cross-correlation method results for noisy output (SNR=0 dB).

			Prediction	Kernel NMSEs	
	α_1	α_2	NMSE (%)	$k_1(m)$ (%)	$k_2(m_1,m_2)$ (%)
LVN-2	0.165	0.811	53.78	7.69	4.10
Cross-correlation	⋯	⋯	86.38	421	1919

to the case when the model output can be decomposed in terms of the hidden unit outputs as

$$y(n)=u_1(n)+\tfrac{1}{2}u_1^2(n)+\tfrac{1}{2}u_2^2(n)+u_3(n)-\tfrac{3}{2}u_3^2(n), \tag{32}$$

where the hidden unit outputs are related to the linear filter outputs of Fig. 2 as $u_1=\nu_1, u_2=\nu_1+\nu_2, u_3=\nu_2$.

The training of the 27 parameters of this network/model is performed using the iterative relations (7)–(9) and (16). The learning curves for the two Laguerre parameters are shown in Fig. 4. The parameters converge to their correct values in the noise-free case after around 1500 iterations (Table 1), although they were both initialized at 0.5.

In the noise-free case, the estimated first-order kernel is shown in Fig. 5, along with the separate slow and fast components. All are almost identical to their true counterparts (see Table 1). The estimated second-order kernel is shown in Fig. 6 along with the separate slow, fast and cross-term (slow-fast) components (all are also almost identical to their true counterparts).

The normalized mean square error (NMSE) of the output prediction achieved by the model (defined as the sum of the squares of the errors between the model prediction and the true output over the sum of the squares of the true output) is on the order of 10^{-3} for the noise-free case. The NMSE is therefore very low and demonstrates the excellent performance of the modeling procedure. Another check of the model accuracy can be performed by the NMSEs of the obtained kernel estimates, which are given in Table 1.

The usefulness of employing two filter banks in modeling systems with fast and slow dynamics can be demonstrated by comparing the performance of the proposed model with that of a LVN with one filter bank having the same complexity (i.e., six LFs, followed by three hidden units with second-degree polynomial activation functions) to model the system. The results are shown in

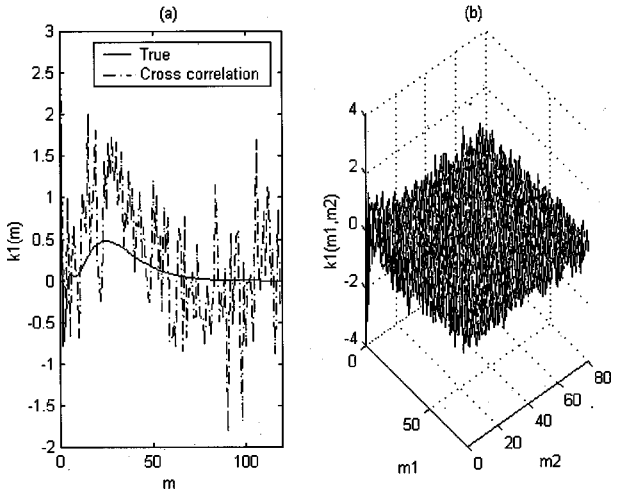

FIGURE 8. Estimated Volterra kernels using the conventional cross correlation technique for the first system, with noisy output (SNR=0 dB): (a) first-order kernel and (b) second-order kernel.

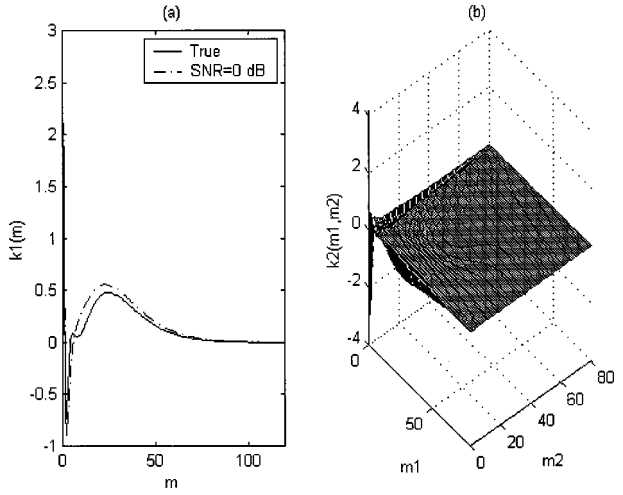

FIGURE 7. Estimated Volterra kernels for the first system, with noisy output (SNR=0 dB): (a) first-order kernel and (b) second-order kernel.

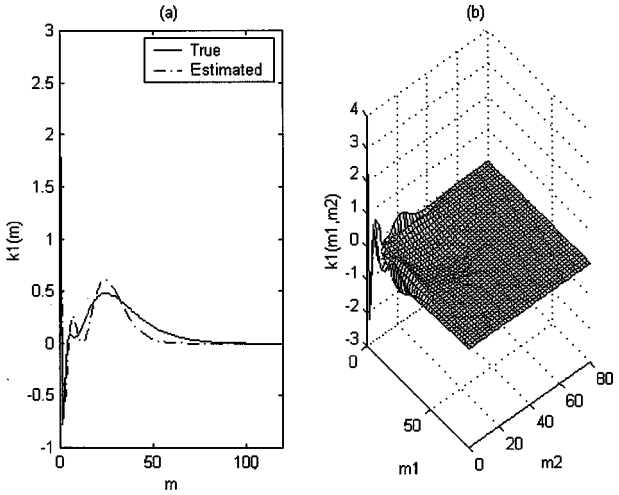

FIGURE 9. Estimated Volterra kernels for the first system using a one filter-bank LVN, noise-free output: (a) first-order kernel and (b) second-order kernel.

TABLE 3. LVN-2 and LVN model performance for the second simulated system; noise-free output.

			Prediction NMSE (%)	Kernel NMSEs	
	α_1	α_2		$k_1(m)$ (%)	$k_2(m_1,m_2)$ (%)
LVN-2	0.056	0.767	0.17	0.10	0.09
LVN	0.443	⋯	0.79	0.62	0.96

Table 1 and Fig. 9 and demonstrate the utility of the proposed method. In order to achieve comparable performance (prediction NMSE of under 1%) with a single filter bank, we have to increase the number of LFs to at least 12, i.e., increase the model complexity considerably by doubling the number of free parameters.

In order to examine the effect of noise on the performance of the model, we add independent GWN to the system output for a signal-to-noise ratio (SNR) equal to 0 dB (i.e., the noise variance equals the mean-square value of the noise-free output). The resulting α learning curves are shown in Fig. 4 in dotted lines. Convergence occurs in about 600 iterations (faster than in the noise-free case) and the estimates of α_1, α_2 are not affected much, as shown in Table 2. The corresponding NMSE value for the model prediction is 53.78%. Since the SNR is equal to 0 dB, the ideal NMSE level should be close to 50%, when the data record tends to an infinite length (since the output mean has a small value). The estimated first- and second-order kernels in the noisy case are shown in Fig. 7 and corroborate the previous conclusion, especially when compared to the kernels obtained via the conventional cross-correlation technique,[4] shown in Fig. 8.

The second system has the same architecture shown in Fig. 2, but with different component definitions. Namely, the linear filter impulse responses are not linear combinations of LFs as before and are given by

$$l_1(m)=\exp\left(-\frac{m}{3}\right)\sin\left(\frac{\pi m}{5}\right), \tag{33}$$

$$l_2(m)=\exp\left(-\frac{m}{20}\right)-\exp\left(-\frac{m}{10}\right). \tag{34}$$

The nonlinearity N is of fourth-order

$$z_1(n)=\nu_1(n)+2\nu_1(n)+4\nu_1^2(n)-4\nu_2^2(n)+4\nu_1(n)\nu_2(n)+\tfrac{1}{3}\nu_1^3(n)+\tfrac{1}{2}\nu_2^3(n)+\tfrac{3}{4}\nu_1^4(n)+\tfrac{1}{2}\nu_2^4(n). \tag{35}$$

As indicated at the end of the Methodology section, we employ an ascending-order search procedure by which the structural parameters of the network/model are selected (i.e., the number of LFs in the filter banks, the number of hidden units, and the degree of the activation functions) using the minimum description length criterion (MDL) based on the mean-square error of the output prediction to terminate the search procedure. In general, the selection of these structural parameters cannot be proven to be unique but a parsimonious model can be obtained using this procedure.

A LVN-2 model with seven LFs in each filter-bank and four hidden units with fourth-degree polynomial activation functions was found to be sufficient (a total of 75 network parameters). The results for a GWN input of 4096 data points are given in Table 3 and demonstrate the excellent performance of the method for this high-order system. The achieved separation of the fast and slow components corresponds to that of the true system. Table 3 also shows the results for a single filter-bank LVN model of the same complexity (i.e., 14 LFs and four hidden units with fourth-degree activation functions).

The effect of output-additive noise on the performance of the model was examined for this system by adding 20 different independent GWN signals to the output, for a SNR of 0 dB. The resulting values of the Laguerre parameters and the NMSEs of the output prediction and the estimated kernels (mean values and standard deviations) are given in Table 4. The robustness of the method is evident since the ideal prediction NMSE is close to 50% and the kernel NMSEs are low compared to the variance of the noisy output data.

Finally, a third system of different structure was studied, described by the following system of differential equations:

$$\frac{dy(t)}{dt}=[-b_0+c_1z_1(t)-c_2z_2(t)]y(t)+y_0, \tag{36}$$

TABLE 4. LVN-2 model performance for the second simulated system for noisy output (SNR=0 dB) using 20 independent runs.

			Kernel NMSEs	
α_1	α_2	Prediction NMSE	$k_1(m)$	$k_2(m_1,m_2)$
0.073±0.031	0.719±0.039	46.12±2.53%	7.05±3.69%	6.03±3.94%

$$\frac{dz_1(t)}{dt} = -b_1 z_1(t) + x(t), \tag{37}$$

$$\frac{dz_2(t)}{dt} = -b_2 z_2(t) + x(t), \tag{38}$$

where $z_1(t)$, $z_2(t)$ are state variables and their products with the output $y(t)$ in Eq. (36) constitute the nonlinearity of this system, which gives rise to an infinite order equivalent Volterra model. The values of the equation parameters are as follows: $b_0=0.5$, $b_1=0.2$, $b_2=0.02$, $c_1=0.3$ and $c_2=0.1$. The contribution of the mth order Volterra term is proportional to the mth powers of $P^{1/2}c_1$ and $P^{1/2}c_2$, where P is the input power level. Since the magnitudes of c_1, c_2 are both smaller than one, a truncated Volterra model can be used to approximate the system. For the earlier values of c_1, c_2, it was found that a fourth-order LVN-2 model was sufficient to model the system. The equivalent Volterra kernels for this system can be analytically derived by using the generalized harmonic balance method.[6] The resulting expressions for the zeroth and first-order kernels are

$$k_0 = \frac{y_0}{b_0}, \tag{39}$$

$$k_1(m) = \frac{y_0}{b_0}\left\{\frac{c_1}{b_1-b_0}[\exp(-b_0 m)-\exp(-b_1 m)] - \frac{c_2}{b_2-b_0}[\exp(-b_0 m)-\exp(-b_2 m)]\right\}. \tag{40}$$

The analytical forms of the higher-order kernels are rather complex and are not given here in the interest of space. The fast component of the first-order kernel corresponds to the first exponential difference in Eq. (40), whereas the slow component corresponds to the second exponential difference. The system was simulated for a GWN input with unity power level and length of 2048 data points by numerical integration of Eqs. (36)–(38), for zero initial conditions. Following the proposed method for the selection of the model order, a LVN-2 with 5 LFs in each filter bank and three hidden units with fourth-degree polynomial activation functions was selected to model the system (a total of 45 free parameters). The obtained results for the noise-free and noisy conditions are given in Table 5, demonstrating the excellent performance of the LVN-2 model for this system as well. It should be noted that the estimated zeroth order kernel was equal to 1.996, very close to its true value of 2, given by Eq. (39).

TABLE 5. LVN-2 model performance for the third simulated system.

	α_1	α_2	Prediction NMSE (%)	Kernel NMSEs $k_1(m)$ (%)
Noise-free output	0.505	0.903	0.28	0.10
Noisy output (SNR=0 dB)	0.366	0.853	47.49	4.13

CONCLUSIONS

The problem of effective modeling of nonlinear systems with fast and slow dynamics was addressed in this paper, since the presence of multiple time scales in the system dynamics may give rise to serious challenges in the modeling task. The combination of Laguerre expansions with feedforward artificial neural networks in the form of the recently introduced Laguerre–Volterra network[1] has been shown to be an efficient way for nonlinear system identification from short input-output records. By employing two different filter banks characterized by distinct Laguerre parameters alpha in the network, the fast and slow system dynamics can be modeled separately by the respective filter-bank. The two Laguerre parameters are estimated on the basis of the data, along with the other model parameters, in a computationally efficient way that ensures that the different time scales of the system dynamics are assigned appropriate alpha values. The effectiveness and robustness of the method in the presence of severe output noise, as well as the advantage of using two filter banks instead of one, were demonstrated by three simulated examples. This raises the possibility of approximating systems that are viewed as having infinite memory with LVN-2 models, whereby one alpha is close to unity, although convergence cannot be guaranteed in such cases and long data records may be required. The application of this methodology to the modeling of dynamic cerebral autoregulation is presented in the companion paper (submitted as Part II).

ACKNOWLEDGMENT

This work was supported by Grant No. RR-01861 awarded to the Biomedical Simulations Resource at the University of Southern California from the National Center for Research Resources of the National Institutes of Health.

REFERENCES

[1] Alataris, K., T. W. Berger, and V. Z. Marmarelis. A novel network for nonlinear modeling of neural systems with arbi-

trary point-process inputs. *Neural Networks* 13:255–266, 2000.
[2] Amorocho, J., and A. Brandstetter. Determination of nonlinear functional response functions in rainfall runoff processes. *Water Resour. Res.* 7:1087–1101, 1971.
[3] Haykin, S. Neural Networks, A Comprehensive Foundation. Upper Saddle, NJ: Prentice-Hall, 1998.
[4] Lee, Y. W., and M. Schetzen. Measurement of the Wiener kernels of a nonlinear system by cross-correlation. *Int. J. Control* 2:237–254, 1965.
[5] Marmarelis, V. Z. Identification of nonlinear biological systems using Laguerre expansions of kernels. *Ann. Biomed. Eng.* 21:573–589, 1993.
[6] Marmarelis, V. Z. Wiener analysis of nonlinear feedback in sensory systems. *Ann. Biomed. Eng.* 19:345–382, 1991.
[7] Marmarelis, V. Z., and X. Zhao. Volterra models and three-layer perceptrons. *IEEE Trans. Neural Netw.* 8:1421–1433, 1997.
[8] Ogura, H. Estimation of Wiener kernels of a nonlinear system and a fast algorithm using digital Laguerre filters. 15th NIBB Conference, Okazaki, Japan, 1985, pp. 14–62.
[9] Rissanen, J. Information theory and neural nets. In: Mathematical Perspectives on Neural Networks, edited by P. Smolensky, M. C. Mozer, and D. E. Rumelhart. Mahwah, NJ: Lawrence Erlbaum, 1996, pp. 567–602.
[10] Watanabe, A., and L. Stark. Kernel method for nonlinear analysis: Identification of a biological control system. *Math. Biosci.* 27:99–108, 1975.