Published in final edited form as:
Ann Biomed Eng. 2009 August ; 37(8): 1654–1667. doi:10.1007/s10439-009-9736-8.

Boolean Modeling of Neural Systems with Point-Process Inputs and Outputs. Part I: Theory and Simulations

Vasilis Z. Marmarelis, **Theodoros P. Zanos**, and **Theodore W. Berger**
Department of Biomedical Engineering, University of Southern California, 1042 Downey Way, DRB 367, Los Angeles, CA 90089-1111, USA

Abstract

This paper presents a new modeling approach for neural systems with point-process (spike) inputs and outputs that utilizes Boolean operators (i.e. modulo 2 multiplication and addition that correspond to the logical AND and OR operations respectively, as well as the AND_NOT logical operation representing inhibitory effects). The form of the employed mathematical models is akin to a "Boolean-Volterra" model that contains the product terms of all relevant input lags in a hierarchical order, where terms of order higher than first represent nonlinear interactions among the various lagged values of each input point-process or among lagged values of various inputs (if multiple inputs exist) as they reflect on the output. The coefficients of this Boolean-Volterra model are also binary variables that indicate the presence or absence of the respective term in each specific model/system. Simulations are used to explore the properties of such models and the feasibility of their accurate estimation from short data-records in the presence of noise (i.e. spurious spikes). The results demonstrate the feasibility of obtaining reliable estimates of such models, with excitatory and inhibitory terms, in the presence of considerable noise (spurious spikes) in the outputs and/or the inputs in a computationally efficient manner. A pilot application of this approach to an actual neural system is presented in the companion paper (Part II).

Keywords

Nonlinear modeling; Spikes; Logical operators

INTRODUCTION

One of the great challenges in the study of the function of the nervous system is the quantitative understanding of the manner in which information is processed by neuronal ensembles in the form of sequences of action potentials (spike-trains or point-processes). Although this problem has been studied for many years, its solution has been possible only in simplified cases/paradigms, because of the intrinsic complexity of this problem arising from the fact that neuronal systems are highly interconnected, nonlinear and dynamic, as well as subject to numerous stochastic influences/variations. An additional challenge is presented by the binary nature of the neuronal signals (spike trains viewed as point-processes) that requires appropriate mathematical methods of analysis and modeling, distinct from the various methods developed for continuous data. We note the fundamental distinction between discretized continuous data (which result from the necessity of sampling imposed by the digital processing of continuous signals) and point-process data (which, by their very nature, contain information only in the timing of their events/spikes). The recent

Address correspondence to Theodoros P. Zanos, Department of Biomedical Engineering, University of Southern California, 1042 Downey Way, DRB 367, Los Angeles, CA 90089-1111, USA. zanos@usc.edu.

availability of multi-unit data recordings through multi-electrode arrays, has added new urgency to the need for practicable modeling methods capable of handling vast amounts of spatio-temporal point-process data in a robust manner.

Although considerable progress has been made over the last 40 years in developing modeling methods appropriate for point-process inputs and outputs that employ trigger-threshold operators, many of these methods are rather simplistic and fail to capture the full complexity of the problem (e.g. the conventional "integrate-and-fire" model).[3,4] Other, more realistic approaches (e.g. Hodgkin-Huxley type of models) become rather complex in their application and eventually impractical, as the number of neuronal units increases.[14,20,21] The recent development and successful application of Volterra-type models in the context of single inputs[1,2,10–13,15–19] and multiple inputs and outputs [5,22–24] offers some promise in this regard, although the complexity of the latter models remains considerable when the number of these inputs and outputs increases. Thus, despite some progress made to date, much remains to be done in order to improve the efficacy of these methods in the context of multi-unit recordings.

This paper introduces a new method of modeling neural systems with point-process inputs and outputs that utilizes Boolean operations (logical AND and OR operations corresponding to modulo-2 multiplication and addition, respectively, as well as the AND_NOT logical operation representing inhibitory effects) in connection with Volterra-type model structures that capture explicitly the hierarchy of multiple interactions among values of the input epoch as they affect the output. Part I presents the underlying theory/methodology and illustrative results from simulations, and Part II presents the corroborating results of an initial application of this methodology to point-process data recorded in the rat hippocampus.

Although considerable mathematical and engineering literature exists on Boolean algebra/operations, especially in the context of digital electronics, no related application to the modeling problem of neuronal point-process transformations is known to us. Some attempts have been made to apply parts of this literature to vision,[6] associative memories[8] and network dynamics.[9] However, the terms *Boolean model* and *Boolean process* are used in those papers with a different meaning and bear no essential relation to the proposed modeling approach.

The proposed methodology of Boolean-Volterra (B-V) modeling is presented in the following section and some illustrative examples from computer simulations are presented in the subsequent section. Although the complexity of the presented examples is limited to the case of two inputs and one output (in the interest of simplicity of presentation), the proposed methodology is readily extendable to the case of multiple inputs and outputs without any modifications. The companion paper (Part II) presents the first application of the proposed methodology to an actual neuronal system (the CA3-CA1 causal pathway in the rat hippocampus).

METHODS

For point-process data, the general mathematical form of the proposed second-order B-V model in the single-input/single-output case is given by the expression:

$$
\begin{aligned}
y(n) = h_1(1) \otimes x(n &- 1) \oplus \cdots \oplus h_1(m) \otimes x(n - m) \varnothing g_1(E_1, I_1) \otimes x(n - m) \otimes x(n - I_1) \ldots \oplus h_1(M) \otimes x(n - M) \varnothing g_1(M, I_M) \otimes x(n - M) \otimes x(n - I_M) \ldots \oplus h_2(1, 2) \otimes x(n - 1) \otimes x(n - 2) \oplus \cdots \oplus h_2(M - 1, \ldots, M) \otimes x(n - M+1) \otimes \cdots \otimes x(n - M)
\end{aligned} \tag{1}
$$

where $y(n)$ and $x(n)$ denote the binary output and input data, respectively, as time-series of 0's and 1's in discrete-time intervals (bins) of *binwidth equal to the refractory period* of the neuron, with 1 indicating the presence of an action potential (spike) at the respective bin. The model coefficients/parameters $h_1(m)$ *and* $h_2(m_1, m_2)$ are binary variables that represent the first-order and second-order excitatory terms of the model respectively. The model coefficients/parameters $g_1(m, Im)$ are binary variables that represent the first-order inhibitory terms of the model, where m denotes the first-order excitatory lag upon which inhibition is applied by the possible presence of an input spike at lag Im. Note that the possible inhibitory terms of the model may only exist in association with excitatory terms and there can be no coincidence of excitatory and inhibitory lags. All these model coefficients/ parameters can be estimated from the input-output data as described below. The Boolean operators $\otimes$ and $\oplus$ denote the modulo-2 operations of multiplication (logical AND) and addition (logical OR) of excitatory action respectively, and the operator Ø denotes the AND_NOT logical operation of inhibitory action. Note that inhibitory terms of second order can also be included in this second-order model form, which can be extended to higher orders to account for higher order interactions.

It is evident in this model form that the terms on the right-hand side (RHS) are structured in a hierarchical manner representing various orders of excitatory and inhibitory interactions among the spikes in the input past epoch as they affect causally the appearance of an output spike at the present time. The first-order terms represent the portion of the model that does not involve nonlinear interactions but describe the causal effect of single spikes in the input past epoch on the appearance of a present output spike. If we define "linear superposition" in the context of systems with point-process inputs and outputs as the lack of higher order excitatory terms and inhibitory terms (i.e. only first-order excitatory effects of the input on the output exist), then the first-order excitatory terms of the RHS of Eq. (1) may be viewed as the "linear" portion of the model. Generally, the set of excitatory/inhibitory terms of *k*th-order may be viewed as the *k*th-order excitatory/inhibitory "Boolean-Volterra (B-V) kernel" of the system, in analogy to Volterra kernels of systems with continuous inputs and outputs.[16]

In each specific neuronal system, only some of the terms on the RHS of Eq. (1) exist (a fact that is expressed mathematically by the respective coefficient being 1). The "system

memory" is defined as the maximum input lag (denoted by M in the model equation) that is present among the terms of the model. The "nonlinear model order" is the largest number of multiplied input lagged values found within any of the existing terms of the model. The finite memory of the system implies mathematically/computationally that the system performs a mapping of a finite binary vector of input past values (equal in length to the system memory and termed the "input past epoch") onto the output present value at each discrete time n.

It is critical to note that the presence of a first-order excitatory term at some lag i eliminates the possibility of having a higher order excitatory term that contains the same lag i, because the nature of the Boolean addition (or logical OR operation) makes each term in the RHS of the model equation *sufficient* for causing an output event. Likewise, the presence of a second-order excitatory term at some pair of lags (i, j) eliminates the possibility of a higher order excitatory term containing this same pair of lags etc. We refer to this fact as the *occlusion effect* and it is expressed mathematically by precluding non-zero values of higher order excitatory B-V kernels at those combinations of lags that contain lags with non-zero values in lower order excitatory B-V kernels. For instance, in the case of a non-zero value of the first-order excitatory B-V kernel at lag i, the second-order excitatory B-V kernel (or higher order excitatory B-V kernels) cannot have non-zero values at pairs of lags (j, k) where *either j or k is equal to i*, because that causal possibility has been taken into account already by the first-order excitatory B-V kernel. Note that the presence of inhibitory terms of p-th order will not alter the occlusion effect in excitatory B-V kernels of higher order. For instance, a first-order inhibitory term $g_1(m, Im)$ occludes the possibility of a second-order excitatory term $h_2(m, Im)$ or a third-order excitatory term $h_3(m, Im, k)$—which are already occluded by the first-order excitatory lag m—but does not occlude possible higher order excitatory terms $h_2(k, Im)$ or $h_3(k, Im, j)$ when k and j are not excitatory lags.

The notion of *sufficiency* of each term on the RHS of the Boolean model leads to the occlusion effect that may have the practical consequence of making the likelihood of non-zero values in higher order B-V kernels smaller with increasing order. For this reason (and in the interest of constraining the notational complexity), we will only present illustrative examples of second-order systems in the following section. However, the mathematical formulation and the applicability of the method extend to higher order systems/models.

Another important observation is that no terms can exist with a multiplicity of lag indices (e.g. the diagonal elements of the second-order excitatory B-V kernel defined by $m_1 = m_2$) because an input spike cannot be considered as interacting with itself. Thus, the diagonal values of all excitatory or inhibitory B-V kernels are set to zero.

The extension of this modeling approach to multiple inputs and outputs is methodologically straightforward and remains computationally efficient, although the notational complexity of the model inevitably increases to accommodate the interactions among the multiple inputs. In the case of multiple outputs, the formulation of the model requires a separate equation for each output of interest expressed in terms of Boolean logical AND, OR and AND_NOT operations of various orders acting on the past values of all inputs (within the epoch of the system memory) that have a causal effect on the present value of the respective output. Consider, for instance, the case of a second-order system with two point-process inputs, $x(n)$ and $u(n)$, and two outputs, $y(n)$ and $z(n)$, that is described by the B-V model equations:

$$
\begin{aligned}
y(n) = {} & h_{yx}(1) \otimes x(n-1) \oplus \cdots \oplus h_{yx}(M) \otimes x(n-M) \\
& \oplus h_{yu}(1) \otimes u(n-1) \oplus \cdots \oplus h_{yu}(M) \otimes u(n-M) \\
& \oplus h_{yxx}(1,2) \otimes x(n-1) \otimes x(n-2) \oplus \cdots \oplus h_{yxx}(m_1,m_2) \otimes x(n-m_1) \otimes x(n-m_2) \oplus \ldots \oplus h_{yxu}(m_1,m_2) \otimes x(n-m_1) \otimes u(n-m_2) \oplus \ldots \oplus h_{yuu}(m_1,m_2) \otimes u(n-m_1) \otimes u(n-m_2) \oplus \ldots \oplus h_{yuu}(M-1,M) \otimes x(n-M+1) \otimes x(n-M)
\end{aligned} \tag{2}
$$

$$
\begin{aligned}
z(n) = {} & h_{zx}(1) \otimes x(n-1) \oplus \cdots \oplus h_{zx}(M) \otimes x(n-M) \\
& \oplus h_{zu}(1) \otimes u(n-1) \oplus \cdots \oplus h_{zu}(M) \otimes u(n-M) \\
& \oplus h_{zxx}(1,2) \otimes x(n-1) \otimes x(n-2) \oplus \cdots \oplus h_{zxx}(m_1,m_2) \otimes x(n-m_1) \otimes x(n-m_2) \oplus \ldots \oplus h_{zxu}(m_1,m_2) \otimes x(n-m_1) \otimes u(n-m_2) \oplus \ldots \oplus h_{zuu}(m_1,m_2) \otimes u(n-m_1) \otimes u(n-m_2) \oplus \ldots \oplus h_{zuu}(M-1,M) \otimes x(n-M+1) \otimes x(n-M)
\end{aligned} \tag{3}
$$

This two-input/two-output model contains no inhibitory terms in the interest of notational simplicity. However, the inclusion of inhibitory terms is possible in a manner similar to the one shown earlier in the model of Eq. (1). Note that each output is represented independently of the other (i.e. only the lagged values of the respective inputs are involved in the terms of the RHS of the model equation for each output). A distinct set of excitatory and inhibitory B-V *self-kernels* exists for each output with respect to each of its inputs. For instance, the second-order excitatory B-V self-kernel of the output $y(n)$ with respect to the input $x(n)$ is $hyxx$. The second-order excitatory interactions between the two inputs with regard to their joint impact on the present value of each output are represented by cross-terms on the RHS of each output equation that involve lagged values of *both* inputs in Boolean product (logical AND operation). Although these interactions are excitatory and of second-order in this example, they can be generally of any order and excitatory or inhibitory (depending on the synaptic configuration within each specific neuronal system). The excitatory/inhibitory cross-terms of each order constitute collectively the excitatory/inhibitory B-V *cross-kernel* of this order. For instance, the second-order excitatory B-V cross-kernel of the output $y(n)$ with respect to the inputs $x(n)$ and $u(n)$ is h_{yxu}. Obviously, there are no cross-kernels of first-order (only self-kernels), because interactions require at least two inputs and, thus, second-order terms at a minimum. The *occlusion effect*, mentioned above in the single-input case, applies also to the B-V *self-kernels and cross-kernels* of the two-input model (or, generally, the B-V kernels of a multi-input model). However, the *diagonal* values of B-V kernels are set to zero *only* for the same input (e.g. self-kernels), but not for distinct inputs (e.g. cross-kernels), because spikes may occur at the same lag (relative to an output spike) for two different inputs.

In the following section, an efficient computational procedure for estimation of the B-V model from input-output point-process data is described. The application of this model estimation procedure is illustrated with simulated data in the Results section and shown to be rather robust in the presence of spurious spikes at the input and/or output (defining "noise" in the point-process context). Note that the estimated B-V model can be used for predicting the point-process output for *any* given input point-process. Because of the binary modality of the output, we cannot use the customary mean-square error of output prediction for model performance evaluation, but we must use instead the relation of true-positive and false-positive predictions of output spikes, as described in the following section.

Boolean-Volterra Model Estimation

The estimation of the B-V model from binary input-output data starts by counting the number of times each possible excitatory term on the RHS of the model (involving input lagged values) *may* have contributed to the generation of an output spike. For instance, in the single-input/single-output case of Eq. (1), we count the lagged input spikes and their Boolean product combinations (as they are defined by the excitatory terms on the RHS of the model equation) that precede each output spike within a memory window of M lags. When this count for each excitatory term is divided by the total number of output spikes, we obtain the *Coincidence Index* (CI) for the respective excitatory kernel lag(s).

The estimation procedure for the B-V model seeks to determine the "active" lags (excitatory or inhibitory) in the various B-V kernels in consecutive steps. The estimation of the excitatory or inhibitory terms requires first the computation of the CI for each lag of the first-order kernel, the inhibitory kernel corresponding to that first-order excitatory kernel, or combination of lags for higher order kernels. These CI values are then rank-ordered (starting with the highest value) for each excitatory kernel. The proposed estimation method starts by considering the lag corresponding to the highest ranked CI value of the highest-order excitatory kernel as a putative "active lag" and computes a "Figure of Merit" (FoM) for the resulting model prediction. This FoM is based on the numbers of "true positives" (NTP) and

"false positives" (NFP) that are computed as correct and incorrect predictions of output spikes respectively. The data-based NTP and NFP values resulting from the application of each putative model are used to compute the FoM according to the expression:

$$\mathrm{FoM}=\log(\mathrm{NTP}) - r\log(\mathrm{NFP}) \tag{4}$$

where r is a weighting-coefficient that defines the relative importance of TPs and FPs in the model prediction. The algorithm proceeds by "activating" the lags with the highest ranked CI values in the lower order kernels and computes the corresponding FoM. If the computed value of the FoM is higher than the previous one, then the respective "activated" lag is accepted as an excitatory "active" lag of the B-V kernel. For each selected excitatory active lag of lower order kernels, the algorithm explores the possible presence of inhibitory lags associated with this excitatory lag by computing the inhibitory CI values based on the false positives of the respective model prediction and calculating the corresponding FoM for the ranked inhibitory lags, as described in the following paragraph. If a larger FoM value results from the activation of an inhibitory lag, then this lag is included in the respective inhibitory B-V kernel. The algorithm proceeds with the successive evaluation of the FoM changes resulting from the inclusion of each new rank-ordered excitatory term on the RHS of the B-V model equation and the possible respective inhibitory terms. When the computed FoM change is positive, then the respective new term (excitatory or inhibitory) becomes part of the B-V kernel. This procedure terminates for each B-V kernel, when a negative change results in the calculated FoM after the activation of a rank-ordered lag (excitatory or inhibitory), yielding the final estimate of the respective excitatory or inhibitory kernel.

The estimation of an inhibitory B-V kernel associated with an excitatory active lag commences with the computation of the CI values of all possible terms of this kernel (corresponding to combinations of all possible input lags with the respective excitatory lag), based on the "false positive" predictions of output spikes resulting from the model composed of all previously estimated B-V kernels. Specifically, we count the lagged input spikes and their Boolean product combinations (as they are defined by the inhibitory terms on the RHS of the model equation) that precede each falsely predicted output spike within a memory window of M lags. This count, divided by the total number of falsely predicted output spikes, represents the CI of the respective inhibitory lag. Next, we compute the changes in the FoM value for each putative model that includes sequentially new rank-ordered inhibitory terms on the RHS of the B-V model equation. The candidate inhibitory term is included in the respective B-V kernel when the resulting FOM value increases, otherwise the candidate term is rejected and the respective inhibitory B-V kernel reaches its final estimate.

The B-V model estimation procedure is terminated when all B-V kernels (both excitatory and inhibitory) reach their "final estimates" that yield the maximum value of FoM. The proposed approach has been found to be rapid and robust, as illustrated in the following section.

Another important issue in the kernel estimation procedure is the aforementioned *occlusion effect* that is due to the fact that a non-zero value of a B-V kernel at some combination of lags prevents the possibility of non-zero values of higher order B-V kernels at those locations that contain the same combination of lags. This fact flows from the sufficiency of each causal possibility represented by the Boolean additive terms (logical OR operation) on the RHS of the model equation. However, as indicated earlier, the presence of inhibitory terms of p-th order may neutralize the occlusion effect in excitatory B-V kernels of order higher than $(p + 1)$.

This procedure can be also used for the selection of the proper model order (i.e. the memory extent M and order Q of the B-V model) by repeating it for successively increasing values of M and Q and quantifying the predictive capability of each model by means of the corresponding value of FoM. The values of M and Q that correspond to the highest FoM value are selected for the final model order. Note that the values of M and Q of each putative model can be extended without affecting the previously computed CI values of the kernels.

A practical issue concerns the selection of the time-binwidth in defining the input-output binary datasets. This selection is dictated by the refractory period of the specific neuronal system and it is recommended to be approximately equal to the refractory period, so that it will not lead to occlusion of existing higher order terms.

RESULTS

As a first illustrative example, consider the second-order single-input/single-output system defined by the B-V kernels shown in Fig. 1. We use a 20 s segment of a Poisson point-process input with mean firing rate (MFR) of 40 sps (spikes per second) to simulate this system and generate the corresponding point-process output that has a MFR of 60 sps in this example. The time-binwidth for the discretization of the data is set at 5 ms, which is viewed as the refractory period of this system. Segments of the input-output data-record are shown in Fig. 2, where the output spikes that are due to the second-order interactions (defined by the second-order excitatory B-V kernel) are highlighted with red, and those canceled by the inhibitory action of the first-order inhibitory B-V kernel are drawn with dashed line. These input-output data sets are used to compute the “Coincidence Indices” (CI) shown in Fig. 3 for the two excitatory and one inhibitory B-V kernels of this system. Using the computed CI values, we subsequently estimate the B-V kernels of the system following the procedure that was outlined in the previous section. The computed FoM values for the tested rank-ordered lags of all kernels are shown in Table 1 for all examples presented herein. The application of the proposed procedure yielded the correct B-V kernels of first and second order (both excitatory and inhibitory) by selecting the actual active lags and only those (no falsely identified active lags). Thus, the B-V model prediction is perfect (i.e. predicting all the actual output spikes and no more). Note that the CI values along the lines $m_1 = 10$ ms and $m_2 = 10$ ms in the second-order kernel are set to zero after the first-order active lag at $m = 10$ ms is selected (occlusion effect).

In order to examine the robustness of the proposed kernel estimation procedure, we repeat the simulation of this system and the estimation of the B-V kernels after we introduce a number of spurious spikes in the input or the output equal to the number of actual spikes in the input or output, respectively (corresponding to 0 dB signal-to-noise ratio). The computed CIs for these two cases are shown in Figs. 4 and 5 for the input-noise and output-noise cases respectively. The resulting FoM values for the tested rank-ordered lags of all kernels are shown in Table 1. Remarkably, the obtained B-V kernel estimates for both the input-noise and output-noise cases are identical to the actual kernels, despite the low signal-to-noise ratio, demonstrating the robustness of this estimation method. Although the kernel estimates are exactly the same as the actual B-V kernels of the system, the final FoM value of the estimated B-V model is not infinite in these cases because of the presence of noise in the input or output. The final FoM values are 3.38 and 6.114 for the input-noise and output-noise cases respectively.

The quality of the output prediction is illustrated in Fig. 6 for the output-noise case, where all the input-related output spikes are predicted correctly, while the unpredicted output spikes are identical to the spurious spikes due to noise. This is, of course, the best performance possible. It should be noted that further increases in the number of spurious

spikes will lead to an increase in the computed CI values of the B-V kernels at the non-active lags and eventually will reach a point where correct identification of the active lags will not be possible. This fact applies universally to all methods of estimation in the presence of noise, although the proposed methodology is shown to be rather robust.

To demonstrate the efficacy of this approach in the two-input case, we consider a second-order B-V system/model with a single output like the one described by Eq. (2), for which the two first-order B-V kernels and the three second-order B-V kernels (two self-kernels and one cross-kernel) are shown in Fig. 7. Note that there are no inhibitory B-V kernels in this example in order to limit the complexity of the presentation. This system is simulated with two independent Poisson input processes with length of 100 s and MFR of 20 sps for each input. The resulting output MFR is 38 sps. There are no spurious spikes in the input or output.

The computed CIs for these kernels are shown in Fig. 8 and attain considerable values even at the nonactive lags, probably due to the presence of the "other" input that acts effectively as "interference" to the reference input. Nonetheless, the application of the proposed estimation procedure yielded the exact B-V kernels, with an infinite value for the final FoM of the estimated B-V model because of the zero prediction error in this noise-free case.

The robustness of the proposed estimation method in the presence of spurious spikes for the two-input case is demonstrated in Fig. 9, where the computed CIs for the five kernels of Fig. 7 are shown for 0 db signal-to-noise ratio in both the input and the output (i.e. the number of spurious spikes in the input *and* the output is equal to the respective number of actual spikes in the input and the output). The resulting B-V kernel estimates are identical to the exact B-V kernels of the system—demonstrating the remarkable robustness of this approach—although the final FoM value of the B-V model is not infinite due to the presence of the spurious spikes.

DISCUSSION

The efficacy of the proposed B-V modeling approach for systems with point-process inputs and outputs has been demonstrated using simulated data in the cases of second-order one-input/one-output and two-input/one-output B-V systems/models. The approach is generally applicable to systems that can be described by model equations of the presented B-V form (including inhibitory components) of any order and with any number of inputs and outputs.

The key to this approach is the hierarchical representation of all causal possibilities (both excitatory and inhibitory) of input binary patterns triggering an output event that is formally encapsulated in the B-V kernels of the model. The estimation of the B-V kernels requires the computation of the CIs and the use of a rank-ordered method of successive evaluations of a Figure-of-Merit (FoM) that quantifies the relative predictive capability of putative models in order to obtain each B-V kernel in a computationally efficient manner. The evaluation of the FoM is based on the numbers of true-positive (NTP) and false-positive (NFP) predictions of output spikes for the respective putative B-V model, taking place in descending rank-order of the data-based CI values of the kernels. Each term that increases the FoM value is incorporated in the respective B-V kernel. This procedure is terminated when the FoM value ceases to improve with the inclusion of a new model/kernel term. This approach can be used for the estimation of both excitatory and inhibitory B-V kernels in two stages (first the excitatory and then the inhibitory kernels, because the latter require estimates of the false-positive output predictions). The efficacy of the proposed approach is premised on the assumption of system stationarity. The proposed B-V kernel estimation method was shown to be computationally efficient and robust in the presence of input and/or output noise.

In the case of input/output noise, the computed CI values at non-active lags are generally larger for increasing noise levels but do not affect the B-V kernel estimation accuracy because their inclusion in a putative model does not increase the FoM (i.e. they do not yield false "active lags" following the proposed methodology). Obviously, as the number of spurious spikes increases (i.e. the noise level increases), there will come a time eventually when it will not be possible to select active lags from the computed CI values even at the actual locations where active lags exist. This corresponds to the case when the noise is overwhelming the actual signal. Although this noise effect is universal to all methods, the proposed approach has been shown to be rather robust (i.e. it takes a lot of noise to reach this undesirable point).

The remarkable robustness of this modeling approach in the presence of spurious spikes in the input and/or output (representing noise in the point-process context) was demonstrated even for cases of severe noise (0 dB signal-to-noise ratio) where the number of spurious spikes equals the number of actual spikes in the input and/or output. No other method known to the authors can demonstrate this degree of robustness in the presence of such heavy point-process noise. Since the spurious spikes are identified by subtraction of the true-positive predicted spikes from the recorded output spikes, the notion of *point-process filtering* emerges in a practical context that may have useful implications (i.e. the removal of spurious spikes in actual multi-unit recordings).

In order to compare with an existing methodology for point-process output systems, the proposed approach was applied and tested with data generated by the original "leaky Integrate-and-Fire (LIF) model" (i.e. a linear filter with an exponential impulse response function followed by a threshold-trigger operator).[7] The resulting B-V model had only a first-order B-V kernel with active lags in its initial portion. The number of these active lags depended on the threshold used in the simulated LIF model (i.e. lower thresholds resulted in more active lags—always in the initial portion of the B-V kernel). No higher order B-V kernels were found to be significant—as expected, since the single-input LIF model does not have intrinsic nonlinear interactions. Other forms of impulse response functions were also tried in the LIF (e.g. difference of two exponentials) and the resulting equivalent B-V model was similar with the previous example, with the only difference being the location of the active lags in the first-order B-V kernel (i.e. the location of the active lags is determined by the location of the high values in the impulse response function).

Although the size of the required input-output data-records for satisfactory estimation of the B-V kernels generally depends on the characteristics of each specific system (i.e. the number and type of terms in its model) and the ambient noise conditions, our initial results from simulated data indicate that a few hundred input-output spikes may be adequate. This suggests that the required length of experimental data-records is likely to be on the order of a few seconds in most cases (i.e. for usual values of input/output MFRs).

This approach can be extended to the multi-input/ multi-output case, since the mathematical formalism and the estimation/evaluation procedures are readily extendable. Of particular importance in the multi-input/multi-output case is the remarkable robustness of this approach in the presence of severe noise (spurious spikes) at the inputs and/or outputs that was demonstrated with simulated data. The key remaining issue is the inevitable rise in complexity of the model (i.e. the number of terms in the model equations) caused by the multitude of possible cross-interactions among the various inputs as they affect each of the outputs.

It is important to note that this modeling approach can also be extended to the auto-regressive case (i.e. when we wish to analyze the internal dynamics of a single point-process

in terms of possible underlying temporal patterns). To this purpose, we may consider the present value of the single process as the putative "output" and we may model its causal relationship with the past epoch of values of the process (extending into the past as long as the "memory" of the process) following the previously outlined procedures and the hierarchical structure of the B-V model. The resulting auto-regressive B-V model is a quantitative representation of the internal dynamics of the point-process and captures possible temporal patterns that may exist consistently within the process. These possible temporal patterns define also the spectral characteristics of the process. As we will demonstrate in the pilot application presented in Part II, this auto-regressive approach may be used to discern and quantify possible "neural rhythms" (e.g. the "theta rhythm" frequently observed in hippocampal activity).

The next critical step is the validation of the proposed modeling approach with actual experimental data. A pilot application to rat hippocampal data is presented in the companion paper (Part II).

Acknowledgments

This work was supported by the NIH/NIBIB grant No. P41-EB001978 to the Biomedical Simulations Resource at USC and by the NSF grant No. EEC-0310723 to the Engineering Research Center for Biomimetic Micro-Electronic Systems at USC.

References

1. Berger TW, Baudry M, Brinton RD, Liaw JS, Marmarelis VZ, Park AY, Sheu BJ, Tanguay AR. Brain-implantable biomimetic electronics as the next era in neural prosthetics. IEEE Proc 2001;89:993–1012.10.1109/5.939806
2. Berger TW, Eriksson JL, Ciarolla DA, Sclabassi RJ. Nonlinear systems analysis of the hippocampal perforant path-dentate system. II. Effects of random train stimulation. J Neurophysiol 1988;60:1077–1094.
3. Burkitt AN. A review of the integrate-and-fire neuron model: I. Homogeneous synaptic input. Biol Cybern 2006;95(2):1–19.10.1007/s00422-006-0068-6 [PubMed: 16622699]
4. Burkitt AN. A review of the integrate-and-fire neuron model: II. Inhomogeneous synaptic input and network properties. Biol Cybern 2006;95(2):97–112.10.1007/s00422-006-0082-8 [PubMed: 16821035]
5. Chen HW, Jacobson LD, Gaska JP. Structural classification of multi-input nonlinear systems. Biol Cybern 1990;63(5):341–357.10.1007/BF00202751 [PubMed: 2223893]
6. Crandall WE. Digital vision theory: Boolean logic model. Int J Neurosci 1991;56(1–4):39–71.10.3109/00207459108985405 [PubMed: 1938146]
7. Dayan, P.; Abbot, LF. Theoretical Neuroscience: Computational and Mathematical Modeling of Neural System. Cambridge: MIT Press; 2001. p. 162
8. D'yachkov AG V, Rykov V. The capacity of the Boolean associative memory. IEEE Intern Conf Artif Neural Netw 1997;1:158–160.
9. Farrow C, Heidel J, Maloney J, Rogers J. Scalar equations for synchronous Boolean networks with biological applications. IEEE Trans Neural Netw 2004;15(2):348–354.10.1109/TNN.2004.824262 [PubMed: 15384528]
10. Gholmieh G, Courellis SH, Fakheri S, Cheung E, Marmarelis VZ, Baudry M, Berger TW. Detection and classification of neurotoxins using a novel short-term plasticity quantification method. Biosens Bioelectron 2003;18(12):1467–1478.10.1016/S0956-5663(03)00120-9 [PubMed: 12941562]
11. Gholmieh G, Courellis SH, Marmarelis VZ, Berger TW. An efficient method for studying short-term plasticity with random impulse train stimuli. J Neurosci Methods 2002;21(2):111–127.10.1016/S0165-0270(02)00164-4 [PubMed: 12468002]

12. Gholmieh G, Courellis SH, Marmarelis VZ, Berger TW. Detecting CA1 short-term plasticity variations with changes in stimulus intensity and extracellular medium composition. Neurocomputing 2005;63:465–481.10.1016/j.neucom.2004.07.001

13. Gholmieh G, Courellis SH, Marmarelis VZ, Berger TW. Nonlinear dynamic model of CA1 short-term plasticity using random impulse train stimulation. Ann Biomed Eng 2007;35(5):847–857.10.1007/s10439-007-9253-6 [PubMed: 17380396]

14. Makarov VA, Panetsos F, de Feoo O. A method for determining neural connectivity and inferring the underlying network dynamics using extracellular spike recordings. J Neurosci Methods 2005;144(2):265–279. [PubMed: 15910987]

15. Marmarelis VZ. Signal transformation and coding in neural systems. IEEE Trans Biomed Eng 1989;36:15–24.10.1109/10.16445 [PubMed: 2646209]

16. Marmarelis VZ. Identification of nonlinear biological systems using Laguerre expansions of kernels. Ann Biomed Eng 1993;21:574–589.

17. Marmarelis, VZ. Nonlinear Dynamic Modeling of Physiological Systems. New York: Wiley Interscience; 2004. p. 359

18. Marmarelis VZ, Berger TW. General methodology for nonlinear modeling of neural systems with Poisson point-process inputs. Math Biosci 2005;196(1):1–13.10.1016/j.mbs.2005.04.002 [PubMed: 15963534]

19. Marmarelis VZ, Orme ME. Modeling of neural systems by use of neuronal modes. IEEE Trans Biomed Eng 1993;40:1149–1158.10.1109/10.245633 [PubMed: 8307599]

20. Meunier C, Segev I. Playing the devil's advocate: Is the Hodgkin-Huxley model useful? Trends Neurosci 2002;25(11):558–563.10.1016/S0166-2236(02)02278-6 [PubMed: 12392930]

21. Richardson MJE. Firing-rate response of linear and nonlinear integrate-and-fire neurons to modulated current-based and conductance-based synaptic drive. Phys Rev E 2007;76:021919-1-15.

22. Song D, Chan RH, Marmarelis VZ, Hampson RE, Deadwyler SA, Berger TW. Nonlinear dynamic modeling of spike train transformations for hippocampalcortical prostheses. IEEE Trans Biomed Eng 2007;54(6):1053–1066.10.1109/TBME.2007.891948 [PubMed: 17554824]

23. Zanos TP, Courellis SH, Berger TW, Hampson RE, Deadwyler SA, Marmarelis VZ. Nonlinear modeling of causal interrelationships in neuronal ensembles. IEEE Trans Neural Syst Rehab Eng 2008;16(4):336–352.10.1109/TNSRE.2008.926716

24. Zanos TP, Courellis SH, Hampson RE, Deadwyler SA, Marmarelis VZ, Berger TW. A multi-input modeling approach to quantify hippocampal nonlinear dynamic transformations. IEEE Eng Med Biol Conf 2006;1:4967–4970.10.1109/IEMBS.2006.260575

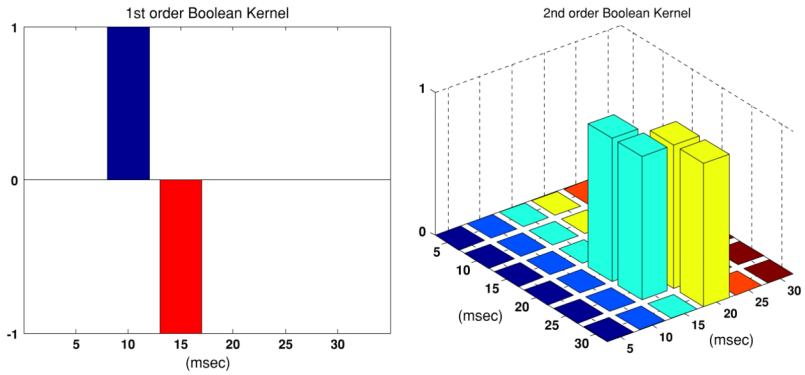

FIGURE 1.

The first-order excitatory and inhibitory (left) and second-order excitatory (right) B-V kernels of the simulated second-order Boolean system/model. The first-order excitatory lag is shown in blue and the associated inhibitory lag in red. Note that the inhibitory lag is strictly associated with the excitatory lag. In the second-order B-V kernel, the identical symmetric values about the diagonal are not shown for clarity of presentation. Note that the diagonal values of the second-order B-V kernel are zero by definition, and its values along the lines at m_1 = 10 ms and m_2 = 10 ms are set to zero due to the occlusion effect (see text).

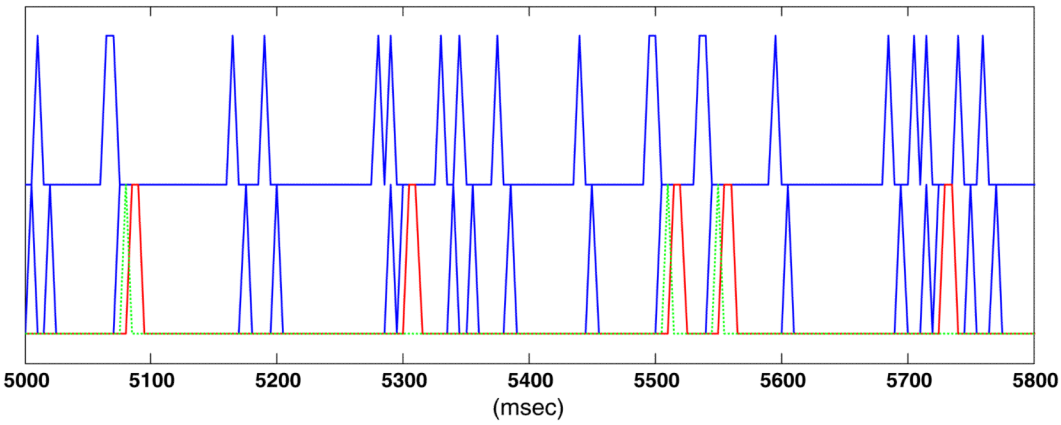

FIGURE 2.

Segments of the simulated input point-process (top trace) and output point-process (bottom trace) for the second-order B-V model defined by the B-V kernels shown in Fig. 1. The output spikes due to the second-order kernel are shown in red and the output spikes suppressed by the inhibitory kernel are shown with green dashed line.

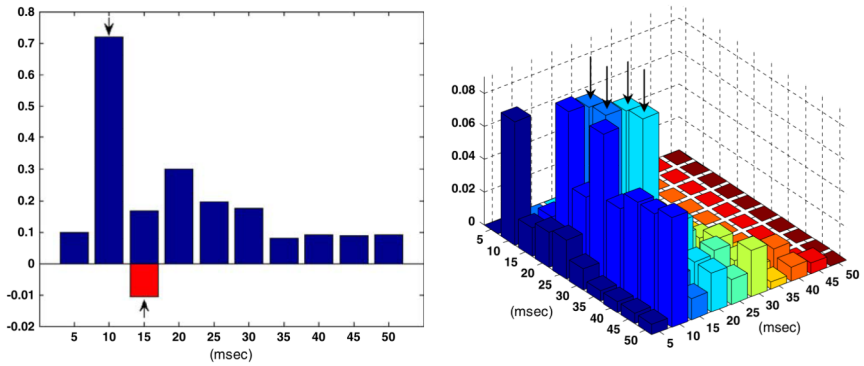

FIGURE 3.

The computed Coincidence Indices (CIs) for the two excitatory and one inhibitory B-V kernels of the simulated system. The “occlusion effect” has not been applied yet. The CI values of the second-order excitatory B-V kernel are only plotted below the diagonal (see text). The black arrows denote the original B-V kernel lags.

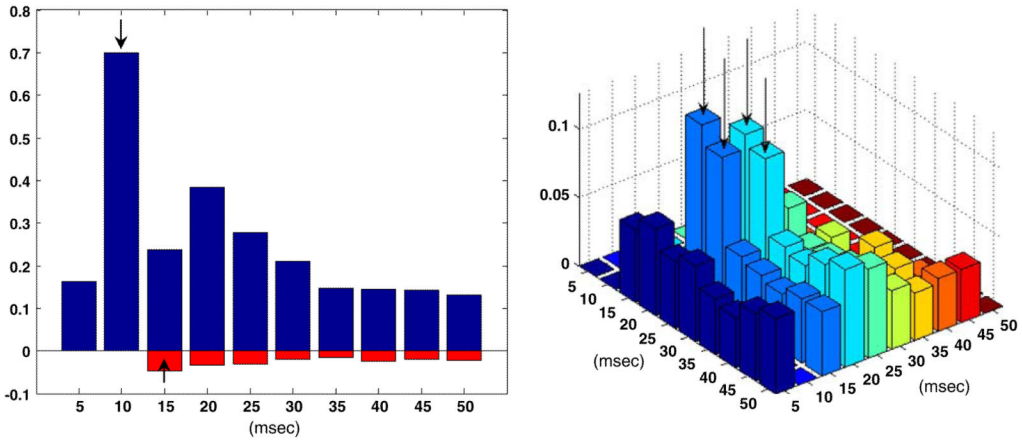

FIGURE 4.

The computed CIs of the B-V kernels in the noisy input case, when the number of spurious spikes is equal to the number of actual spikes in the input. The black arrows denote the original B-V kernel lags.

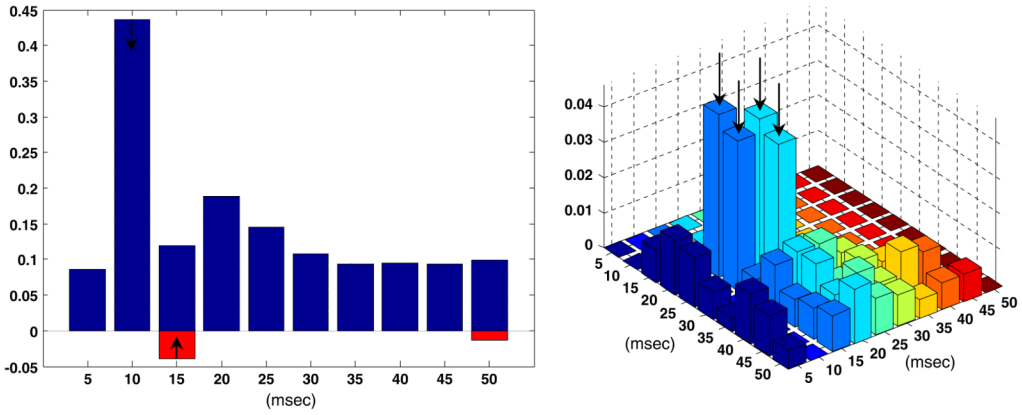

FIGURE 5.

The computed CIs of the B-V kernels in the noisy output case, when the number of spurious spikes is equal to the number of actual spikes in the output. The black arrows denote the original B-V kernel lags.

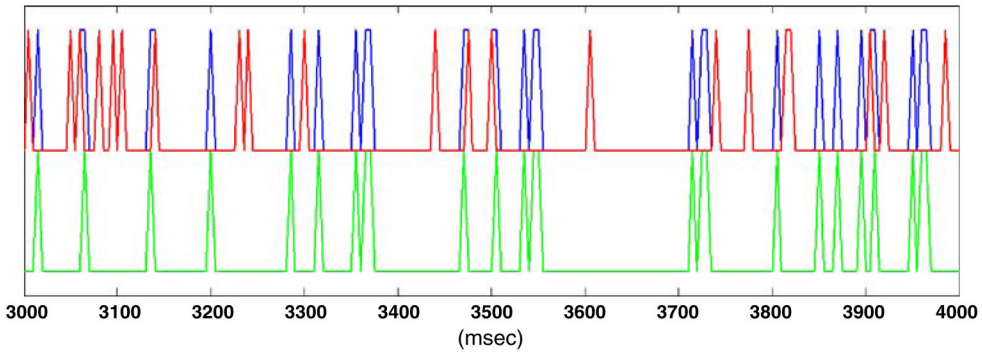

FIGURE 6.
An illustration of the quality of the B-V model prediction (green spikes) in the output-noise case. All the input-dependent output spikes (shown in blue) are correctly predicted and the unpredicted output spikes are identical to the spurious spikes due to noise (shown in red).

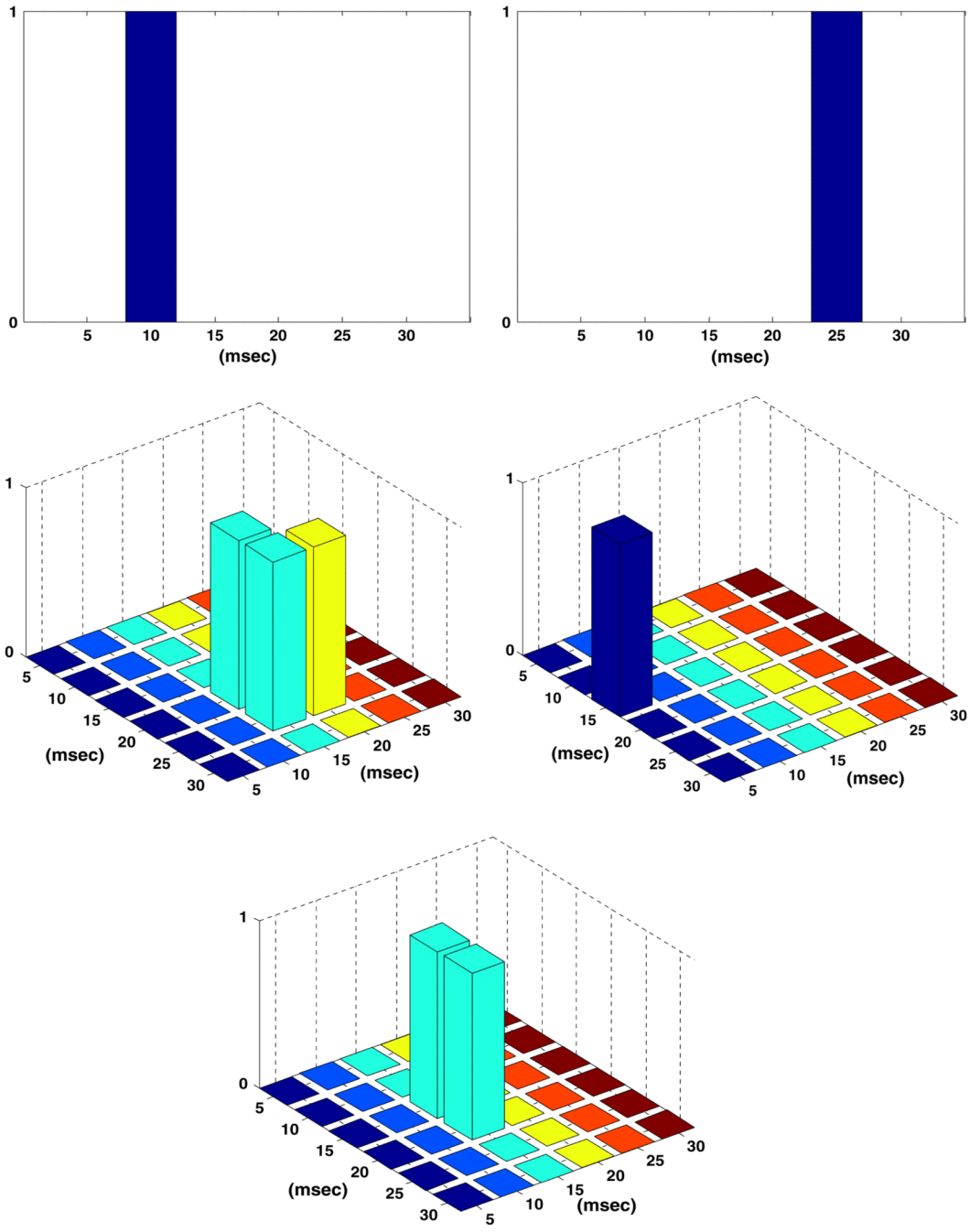

FIGURE 7.

The two first-order (top) and three second-order (bottom) excitatory B-V kernels of the simulated B-V system/model with two-inputs and one output. The cross-kernel (shown at the bottom) has a non-zero value at a diagonal point, unlike the self-kernels that cannot.

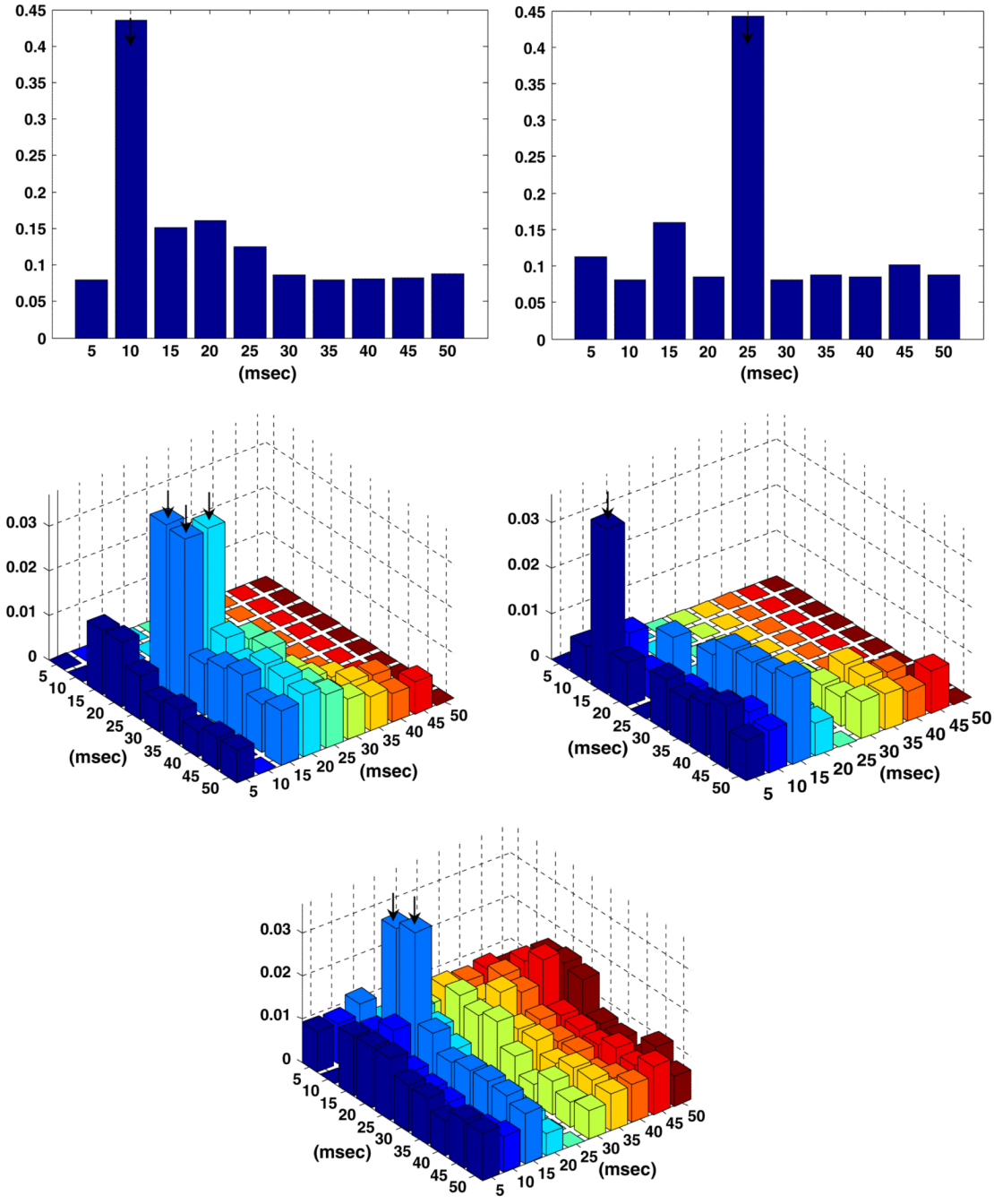

FIGURE 8.

The computed CIs for the five B-V kernels of Fig. 7 in the noise-free case of the simulated system/model with two inputs and one output. Note that the CIs of the cross-kernel are computed for the entire domain, including the diagonal, unlike the self-kernels where the CIs are computed only for the triangular domain below the diagonal (see text). The black arrows denote the original B-V kernel lags.

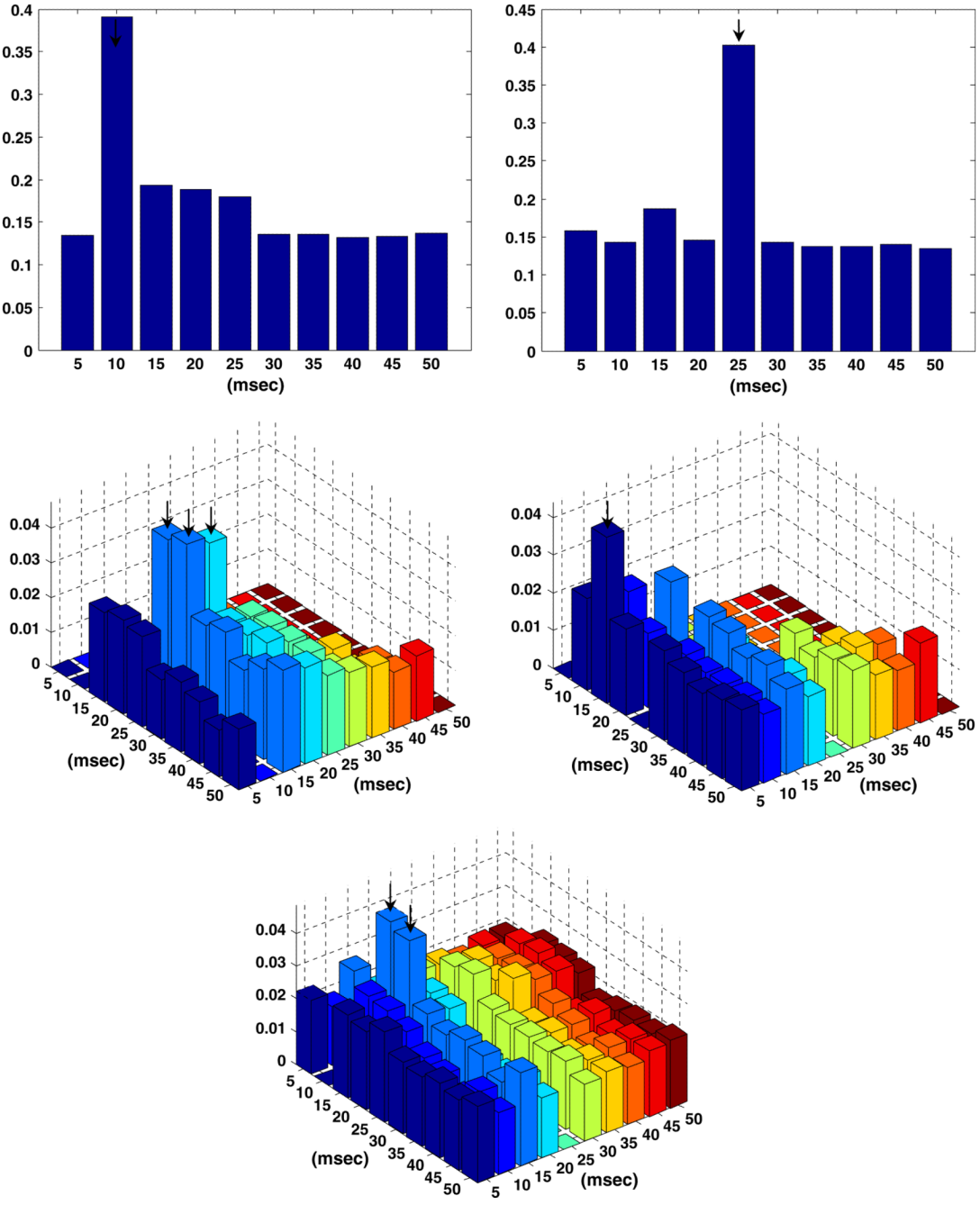

FIGURE 9.

The computed CIs for the five B-V kernels of Fig. 7 for the two-input/single-output system/model in the case of 0 dB signal-to-noise ratio in the input and output. The computed CI values are somewhat elevated (relative to the noise-free case) due to the spurious spikes. The black arrows denote the original B-V kernel lags.

TABLE 1

The computed FoM values for all the lags in the presented examples.

Step	Lag added	FoM (r = 0.5)
Noise-free case		
1	k2(4,2)	3.73
2	k1(1,2)	4.1
3	ki(2,3)	5.79
4	ki(2,1)	5.79 (stop ki)
5	k2(5,3)	5.9108
6	k1(1,4)	3.423 (stop k1)
7	k2(6,4)	6.0064
8	k2(4,3)	6.0799
9	k2(5,4)	Infinite
10	k2(8,4)	4.494 (algorithm stops)
Input noise case		
1	k2(4,2)	2.753
2	k1(1,2)	3.211
3	ki(2,3)	3.267
4	ki(2,8)	3.205 (stop ki)
5	k2(6,4)	3.319
6	k1(1,4)	3.012 (stop k1)
7	k2(5,4)	3.347
8	k2(4,3)	3.368
9	k2(5,3)	3.38
10	k2(4,1)	3.325 (algorithm stops)
Output noise case		
1	k2(10,2)	2.944
2	k1(1,2)	4.16
3	ki(2,3)	5.802
4	ki(2,10)	5.68 (stop ki)
5	k2(4,3)	5.905
6	k1(1,4)	3.404 (stop k1)
7	k2(5,4)	5.987
8	k2(5,3)	6.059
9	k2(6,4)	6.114
10	k2(4,1)	4.673 (algorithm stops)
2-Inputs noise free		
1	k2ab(2,3)	5.192956851
2	k2a(6,2)	5.726847748
3	k2b(5,1)	6.154858094
4	k1a(1,2)	7.493873887
5	k1b(1,5)	8.065579427

Step	Lag added	FoM (r = 0.5)
6	k2ab(4,3)	8.100464891
7	k2a(4,3)	8.131530711
8	k2b(3,1)	8.161089513
9	k1a(1,4)	4.71607475 (stop k1a)
10	k1b(1,3)	4.69969291 (stop k1b)
11	k2ab(3,3)	8.186464429
12	k2a(5,4)	8.210939733
13	k2b(10,3)	6.03903702 (stop k2b)
14	k2ab(4,9)	5.9779856 (stop k2ab)
15	k2a(5,3)	8.235095497
16	k2a(6,4)	6.044 (algorithm stops)
2-Inputs noisy case		
1	k2ab(2,5)	4.510577536
2	k2a(4,2)	4.74071208
3	k2b(5,9)	4.895411181
4	k1a(1,2)	5.34838321
5	k1b(1,5)	5.658714382
6	k2ab(3,3)	5.671893473
7	k2a(5,3)	5.681807742
8	k2b(3,1)	5.690428702
9	k1a(1,3)	5.527971156 (stop k1a)
10	k1b(1,3)	5.52347704 (stop k1b)
11	k2ab(4,3)	5.690982416
12	k2a(4,3)	5.693455815
13	k2b(4,3)	5.66856623 (stop k2b)
14	k2ab(4,6)	5.6661622 (stop k2ab)
15	k2a(5,4)	5.697013806
16	k2a(7,3)	5.671 (algorithm stops)

Note that the occlusion effect is not applied at the first step of the algorithm, since no active first-order lags have been selected yet.