# AN EXTENSION OF THE M–SEQUENCE TECHNIQUE FOR THE ANALYSIS OF MULTI–INPUT NONLINEAR SYSTEMS

Ethan A. Benardete and Jonathan D. Victor

Laboratory of Biophysics
The Rockefeller University
and
Department of Neurology and Neuroscience
Cornell University Medical College

## ABSTRACT

*White–noise analysis and related methods of nonlinear systems identification describe a physical system's response to its input in terms of "kernels" of progressively higher orders. A popular analytic scheme in the laboratory uses a class of pseudorandom binary sequences, m–sequences, as a test signal. The m–sequence method has several advantages for investigating linear and nonlinear systems: ease of implementation, rapid calculation of system kernels, and a solid theoretical framework. One difficulty with this method for nonlinear analysis comes from the algebraic structure of m–sequences: linear and nonlinear terms can be confounded, especially in the analysis of systems with many inputs. We have developed a modification of the m–sequence method which allows control of these anomalies. This method is based on input signals consisting of a superposition of m–sequences whose lengths are relatively prime. The fast computational methods which facilitate kernel calculation for a single m–sequence input are readily extended to this new setting. We describe the theoretical foundation of this method and present an application to the study of ganglion cells of the macaque retina.*

## INTRODUCTION

Nonlinear systems analysis has widespread applications in biology. One of the most general approaches is Wiener white–noise analysis (Marmarelis and Marmarelis, 1978). Standard Wiener analysis in the time domain, based on the presentation of white noise to the system under study and dissection of the resulting response, has theoretical advantages (Wiener, 1958), but a straightforward application of the Wiener method is compromised by the impossibility of generating true white–noise in the laboratory. For this reason, variations of the Wiener procedure, which share its theoretical framework but rely on alternative input signals, have been developed. These variations include frequency–domain methods (e.g., the sum of sinusoids technique (Victor and Knight, 1979)) and time–domain methods (e.g., band–limited Gaussian white–noise (Marmarelis and Naka, 1972)). The Wiener and related methods derive their utility from being able to separate a system's response

*Advanced Methods of Physiological System Modeling, Vol. 3,*
Edited by V.Z. Marmarelis, Plenum Press, New York, 1994

into "kernels," each of which is a component of a stereotyped mathematical description of the system's behavior. The order of the kernel function describes how many input values are simultaneously multiplied, weighted by that kernel, and summed into the total response. The fundamental kernel representation of a system is that of Volterra (1932). The response, $r(t)$, of a single–input system to an input, $s(t)$, has a representation in terms of Volterra kernels which begins:

$$r(t) = L_0 + \int_0^\infty L_1(\tau)s(t-\tau)d\tau + \int_0^\infty \int_0^\infty L_2(\tau_1,\tau_1)s(t-\tau_1)s(t-\tau_2)d\tau_1 d\tau_2 +$$

$$\int_0^\infty \int_0^\infty \int_0^\infty L_3(\tau_1,\tau_2,\tau_3)s(t-\tau_1)s(t-\tau_2)s(t-\tau_3)d\tau_1 d\tau_2 d\tau_3 + \ldots \quad (1)$$

Here, $L_0$ is the zeroth–order kernel, $L_1(\tau)$ is the first–order kernel, $L_2(\tau_1,\tau_2)$ is the second–order kernel, and $L_3(\tau_1,\tau_2,\tau_3)$ is the third–order kernel. The zeroth–order kernel describes the response of the system to no input. The first–order kernel function attaches a weight to values of the input at specific times in the past according to how much an input value affects the total response function. For a linear system, the first–order kernel is known as the impulse response function. The second–order kernel assigns weights to the product of two previous input values. The third–order kernel assigns weights to the product of three previous input values and so on. The task for the experimenter is to determine the system kernels in an efficient, practical, and sufficiently complete manner.

Among different system identification procedures, frequency-domain methods typically allow clean separation of nonlinear kernels of various orders, but provide only a limited number of data points on each kernel. Time-domain methods (including m–sequence methods), by comparison, provide more data points on each kernel, but can confound linear and nonlinear responses. The goal of this paper is to provide an improved m–sequence method for multi–input nonlinear systems.

We will initially focus on the statistical and algebraic properties of m–sequences underlying both the advantages and the disadvantages of their use in kernel measurement. Next, we will present the hybrid method which allows the experimenter to balance improved separation of nonlinear terms with a reasonable number of data points per kernel. Finally, we will discuss how this method differs from previous methods, and briefly present an application of this technique.

## Preliminaries

It is necessary to make some approximations and assumptions in order to estimate kernels experimentally. First, in the laboratory, the system's response is sampled at discrete time intervals, not continuously. The first measurement is usually considered to take place at time $t = 0$; the next at $t = \Delta T$, and the next at $t = 2\Delta T$, and so on, where $\Delta T$ is the sampling interval. Therefore, it is natural to expect that kernel values will only be known at time lags which are a multiple of the sampling interval, $\Delta T$, and to index kernel values by this multiple. For convenience, we can assume that the unit of time has been chosen so that $\Delta T = 1$.

Secondly, we assume that the system's kernels are of finite duration. In other words, an input at the present time will only affect the system's response for a limited amount of time in the future, not indefinitely. This length of time is generally referred to as the system's memory, and will be denoted by $T$. This memory may be determined from *a priori* knowledge of the system, or from pilot studies based

on systems-analytic methods. Knowledge of the system's memory is important in designing a protocol to carry out a kernel measurement (see below). Furthermore, the only systems we will consider are those with kernels that are functions solely of past and present input, i.e., causal.

Finally, we assume that the system's kernels do not vary over the course of the experimental measurement. This system property is referred to as time-stationarity.

With this preface in mind, we take another look at the system described by Eq. (1). Under the preceding assumptions, the integrals become summations indexed by the sampling interval:

$$r(t) = L_0 + \sum_{k=0}^{T} L_1(k)s(t-k) + \sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)s(t-k_1)s(t-k_2) +$$

$$\sum_{k_1=0}^{T}\sum_{k_2=0}^{T}\sum_{k_3=0}^{T} L_3(k_1,k_2,k_3)s(t-k_1)s(t-k_2)s(t-k_3) + \ldots \qquad (2)$$

where $L_0$, $L_1(k)$, $L_2(k_1,k_2)$, and $L_3(k_1,k_2,k_3)$ are discrete versions of the kernels in Eq. (1) and $T$ is the system's memory in discretized time units.

### Types of Kernels

Having chosen to identify a physical system in terms of a functional expansion, it remains to determine the kernel functions that describe the properties of the system. To justify our efforts, the functional expansion that we derive should meet certain requirements.

One goal of modeling is to produce a mathematical formulation that will predict the physical system's behavior under certain conditions. A measurement of how well the model mimics the actual system is the mean-squared error (MSE): the average of the square of the difference between the model's prediction and the physical system's output. Therefore, a natural requirement for a functional model is that it minimizes this error in some sense. Secondly, since polynomial functions may be poor approximations to biological nonlinearities (Victor and Shapley, 1979), a kernel expansion should handle a variety of nonlinearities that may or may not be "analytic" in a mathematical sense.

The Volterra formalism introduced in Eq. (1) cannot meet these requirements. First, the Volterra kernels do not meet the MSE criteria. For example, a cubic nonlinearity lacks a Volterra representation in the kernels of less than third–order. The first- and second- order kernels ($L_1 = 0$ and $L_2 = 0$) do not comprise the best-fitting first- and second–order models. In addition, measuring the Volterra kernels is experimentally very difficult if not impossible (Victor, 1992), especially for biological systems with sharp nonlinearities.

The Wiener functional expansion, on the other hand, is based on the system's response to white–noise of a certain power (variance). The kernels are constructed such that the $n$th-order Wiener expansion of a system minimizes the difference between the model's response and the measured response to the white–noise input upon which the expansion is based. The Wiener expansion thus depends on the power of the white–noise input; Volterra series expansions are based on responses to infinitesimal inputs.

For analytic systems, the Volterra series representation can be rearranged into a Wiener representation, and vice-versa. Wiener and Volterra kernels are not identical.

Wiener kernels of a given order, $n$, contain contributions from Volterra kernels not only of order $n$, but also of certain higher–orders $(n+2, n+4, n+6, ...)$. These contributions from Volterra kernels of higher–order correspond to terms such as $L_4(k_1, k_2, k, k)$, where an even number of time lags are equal, i.e., the "diagonal" elements.

For non-analytic systems which do not have a Volterra representation, the Wiener representation typically exists, and in principle can be measured (Schetzen, 1980). Furthermore, the Wiener kernels, once measured, provide a sound avenue for developing or verifying the validity of a variety of models of biological systems. (For a brief theoretical account of Wiener kernels, see Appendix I). In the following analysis, the goal of the hybrid m–sequence method will be to estimate the Wiener kernels of a system of interest.

### M–Sequences

Below, those properties of m–sequences that are needed for the hybrid m–sequence method are summarized. A more extensive discussion of properties of m–sequences and procedures for generating them may be found elsewhere (Golomb, 1968; Gyftopoulos and Hooper, 1964; Barker and Pradisthayon, 1970; Ream, 1970; Sutter, 1987).

A binary m–sequence of order $N$ is a cyclic sequence $\{b_k\}$ of 0's and 1's that satisfies a linear recurrence relation:

$$b_k \equiv \sum_{l=1}^{N} b_{k-l} r_l \tag{3}$$

where $\{r_1, \ldots, r_N\}$ are the coefficients (either 0 or 1) of the recurrence relation, and $\equiv$ denotes congruence (mod 2). An m–sequence has length: $M = 2^N - 1$. For use as a test signal,

$$m(k) \equiv 1 - 2b_k. \tag{4}$$

For example, an m–sequence of length $(2^3 - 1)$ is the binary sequence $\{1,0,0,1,0,1,1\}$ which corresponds to the sequence $\{-1,1,1,-1,1,-1,-1\}$ for $m(k)$. Thus, $m(1) = -1$, $m(2) = 1, ..., m(7) = -1$. A time-shift of the m–sequence generates another m–sequence, which starts at a new initial position. The above sequence shifted by +5 reads $m(k+5) = \{-1,-1,-1,1,1,-1,1\}$. These shifts exploit the cyclic nature of the sequence. In what follows, we write the sequence as a function of the variable $t$, to indicate that it takes on a new value (1 or –1) after each unit of discrete time.

M–sequences satisfy certain properties that make them especially useful for systems analysis (Golomb, 1968). Three of these are frequently needed and so are stated here. First,

$$\langle m(t) \rangle = -\frac{1}{M} \tag{5}$$

where $\langle\rangle$ denotes the average over complete cycles of the m–sequence. That is, the sum of all the elements of an m–sequence divided by the length of the sequence is $-\frac{1}{M}$, where $M$ is the length of the sequence. This property is based on the fact that every m–sequence has one more –1 than +1 in it.

For our purposes, it is also necessary to know how to compute the product of two shifts of the same m–sequence (multiplied together element by element). The second property gives this result:

$$m(t+a)\cdot m(t+b) = m(t+F(a,b)) \tag{6}$$

provided that $a \neq b$. (If $a = b$, then $m(t+a)\cdot m(t+b) = 1$, since $m(t)$ is always +1 or –1). Eq. (6) states that the product of two distinct shifts of an m–sequence is a third shift of the same sequence. This shift is determined by a mapping, $F$, which depends on the choice of $r$'s in the recurrence relation of Eq. (3). To illustrate with the first example, $m(t)\cdot m(t+5) = \{1,-1,-1,-1,1,1,-1\} = m(t+4)$, thus $F(0,5) = 4$.

A third useful property, which is typically called the autocorrelation property of m–sequences, follows from the two above:

$$\langle m(t+a)\cdot m(t+b)\rangle = \begin{cases} 1 & \text{if } a = b \\ -\frac{1}{M} & \text{otherwise} \end{cases} \tag{7}$$

In Eq. (7), if the two shifts, $a$ and $b$, are not equal, multiplication of the shifted m–sequences produces a third, the average of which is $-\frac{1}{M}$ as in Eq. (5). If $a = b$, multiplication produces a sequence of all 1's. The autocorrelation property establishes the m–sequences as a class of pseudorandom, quasi-white signals.

The ensemble average of the product of two signals may be considered to be an inner product. In this sense, two non-identical shifts of the same m–sequence are nearly orthogonal since their inner product is small ($-\frac{1}{M}$ ). However, a shift of an m–sequence is not necessarily orthogonal to a product of two other shifts:

$$\langle m(t+a)\cdot m(t+b)\cdot m(t+c)\rangle = \begin{cases} 1 & \text{if } a = F(b,c) \text{ or } b = F(c,a) \text{ or } c = F(a,b) \\ -\frac{1}{M} & \text{otherwise} \end{cases} \tag{8}$$

These higher–order correlations are the source of the difficulty in the estimation of higher–order Wiener kernels via m–sequence inputs (Barker and Pradisthayon, 1970).

### Standard Approach

Lee and Schetzen (1965) showed that the Wiener kernels of a nonlinear system could be estimated by cross–correlating the response of the system with the Gaussian white–noise input. For certain m–sequence inputs the Wiener kernels can also be approximated by cross–correlating the response of a system with the m–sequence test signal (Gyftopoulos and Hopper, 1964; Ream, 1970; Sutter, 1987). Sutter (1991) has also developed a fast transform method for this cross–correlation, the Fast M Transform (FMT). The cross–correlation algorithm for the first few Wiener kernels is:

$$h_0 = \langle r(t)\rangle \tag{9}$$

$$h_1(l) = \langle r(t)\cdot s(t-l)\rangle \tag{10}$$

$$h_2(l_1,l_2) = \frac{1}{2!}\langle r(t)\cdot s(t-l_1)\cdot s(t-l_2)\rangle, \text{ for } l_1 \neq l_2 \tag{11}$$

$$h_3(l_1, l_2, l_3) = \frac{1}{3!}\langle r(t) \cdot s(t - l_1) \cdot s(t - l_2) \cdot s(t - l_3)\rangle, \text{for } l_1 \neq l_2 \neq l_3 \tag{12}$$

where $h_0$, $h_1$, $h_2$, and $h_3$ are the zeroth-, first-, second-, and third–order Wiener kernels, and $\langle\rangle$ indicates an average over all values of the (unit power level) stimulus.

Consider the system whose input-output relationship is given by Eq. (2) truncated after second–order. The response of this system to the m–sequence $m_1(t)$ will be:

$$r(t) = L_0 + \sum_{k=0}^{T} L_1(k)m_1(t-k) + \sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1, k_2)m_1(t-k_1)m_1(t-k_2) \tag{13}$$

Now we will use the Eqs. (9–12) to measure the kernels $h_0$ through $h_2$, with $m_1(t)$ playing the role of $s(t)$. We will denote these kernel estimates by $\hat{h}_0^1, \hat{h}_1^1, \hat{h}_2^1, \ldots \hat{h}_n^1$ to indicate that the estimates are derived from the cross–correlation algorithm (9) through (12) with an m–sequence, $m_1(t)$, as input, rather than with a Gaussian white-noise input. From Eq. (9) and the above properties of m–sequences, the resulting estimate of $h_0$ is:

$$\hat{h}_0^1 = L_0 - \frac{1}{M}\sum_{k=0}^{T} L_1(k) + \sum_{k=0}^{T} L_2(k,k) - \frac{1}{M}\sum_{k_1 \neq k_2}^{T} L_2(k_1, k_2) \tag{14}$$

Eq. (14) shows that an average of the response produces an estimate, $\hat{h}_0^1$, which has three kinds of contributions. First, there is a contribution from $L_0$, which indicates the response of the system to zero input. Second, there is a small contribution due to the imperfect "randomness" of the m–sequence, whose size is inversely proportional to the length of the m–sequence. We denote the size of this contribution by $O(\frac{1}{M})$ to indicate that it approaches zero with the same rapidity as $\frac{1}{M}$, i.e., presenting the system with a longer m–sequence can further limit this contribution. With this notation, Eq. (14) takes the more compact form:

$$\hat{h}_0^1 = L_0 + O(\frac{1}{M}) + \sum_{k=0}^{T} L_2(k,k) \tag{15}$$

The third term in Eq. (15) is a contribution of the second–order Volterra kernel, $L_2$, to the zeroth–order kernel estimate. This term is present in both the standard Wiener kernel $h_0$ and the m–sequence estimate $\hat{h}_0^1$. It is a consequence (see above) of the relationship between the Volterra kernels and the Wiener kernels.

Now let us examine the estimate $\hat{h}_1^1(l)$ derived from the cross–correlation algorithm in Eq. (10) with $m_1(t)$ playing the role of $s(t)$. In order for $\hat{h}_1^1(l)$ to be a reasonable estimate of $h_1(l)$, the length, $M$, of the m–sequence should exceed the memory of the system, $T$; otherwise, $\hat{h}_1^1(l)$ has contributions from $L_1(l)$ and other points on that kernel separated by the length of m–sequence. In what follows, we shall assume that a reasonable estimate of the system's memory has been made such that an m–sequence of sufficient length ($> T$) can be selected.

For the system in Eq. (13), we apply the autocorrelation properties of m–sequences given by Eq. (7) and Eq. (8) to obtain:

$$\neq l_2 \neq l_3 \qquad (12)$$

ırder Wiener kernels,
vel) stimulus.
iven by Eq. (2) trun-
-sequence $m_1(t)$ will

$$\text{ı})m_1(t-k_2) \qquad (13)$$

ırough $h_2$, with $m_1(t)$
es by $\hat{h}_0^1, \hat{h}_1^1, \hat{h}_2^1, \ldots \hat{h}_n^1$
elation algorithm (9)
ith a Gaussian white-
uences, the resulting

$$\text{ı}, k_2) \qquad (14)$$

imate, $\hat{h}_0^1$, which has
ı $L_0$, which indicates
ll contribution due to
ıversely proportional
ontribution by $O(\frac{1}{M})$
s $\frac{1}{M}$, i.e., presenting
ntribution. With this

(15)

ɔrder Volterra kernel,
in both the standard
quence (see above) of
ernels.
cross–correlation al-
rder for $\hat{h}_1^1(l)$ to be a
ce should exceed the
from $L_1(l)$ and other
In what follows, we
y has been made such

ɔn properties of m–

$$\hat{h}_1^1(l) = L_1(l) + O(\frac{1}{M}) + \sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)\delta(l, F_1(k_1,k_2)) \qquad (16)$$

where $\hat{}1$ indicates the dependence of the estimate on the sequence $m_1(t)$, and where $\delta(l,k) = 1$ if $l = k$ and 0 otherwise. The above expression shows that $\hat{h}_1^1$ includes $L_1$ and an $O(\frac{1}{M})$ part, but also potential contributions from the second–order kernel, $L_2$. These contributions to $\hat{h}_1^1(l)$ are not necessarily small. They depend on the size of the second- order kernel, $L_2(k_1,k_2)$, at the point at which $F_1(k_1,k_2) = l$. ($F_1$ is the mapping guaranteed by Eq. (6) to take the product of two shifts of $m_1(t)$ to a single shift of $m_1(t)$). These overlap contributions of the second–order kernel to the first–order kernel estimate do not correspond to Wiener/Volterra interrelationships, but rather are anomalies which are due solely to the algebraic properties of the m–sequence (Barker and Pradisthayon, 1970; Ream, 1970). The removal of these anomalies is the main benefit of the hybrid method.

The estimate $\hat{h}_2^1$ of $h_2$ is the cross–correlation of $r(t)$ with two shifts of the stimulus, as indicated in Eq. (11). Along the diagonal, the fact that the input signal can only be $\pm 1$ implies

$$\hat{h}_2^1(l,l) = \frac{1}{2}h_0 \qquad (17)$$

Thus, this algorithm provides no information for points on the diagonal of $h_2$, the second–order kernel.

Off the diagonal, we find from Eq. (11) that

$$\hat{h}_2^1(l_1,l_2) = L_2(l_1,l_2) + O(\frac{1}{M}) + \frac{1}{2}\sum_{k=0}^{T} L_1(k)\delta(k, F_1(l_1,l_2)) +$$

$$\frac{1}{2}\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)\delta(F_1(k_1,k_2), F_1(l_1,l_2)) \qquad (18)$$

The estimate of $\hat{h}_2^1(l_1,l_2)$ has contributions from $L_2(l_1,l_2)$ and an $O(\frac{1}{M})$ part, but also contributions from $L_1$ and other points of $L_2$, as determined by the mapping $F_1$. These "overlaps" of first–order and second–order kernels amount to errors in the estimation of the Wiener kernel, again due to the algebraic properties of m–sequences.

Thus, estimates of kernels obtained by cross–correlation of responses with m–sequence inputs differ from Volterra kernels in three ways. One disparity corresponds to the slight difference in the number of +1's and –1's in the m–sequence. This is $O(\frac{1}{M})$, where $M$ is the length of the m–sequence (typically > 1000), and will be ignored from now on. The second difference corresponds to the difference between Wiener kernels and Volterra kernels in general, i.e., the "diagonal" terms discussed above. These are terms which are required by the fundamental differences between orthogonal (Wiener) series and power (Volterra) series. Finally, there are terms which relate specifically to anomalies in the higher–order statistics of m–sequences (i.e., triple or higher order auto–correlations). These terms, which depend on the choice of m–sequence through the mapping $F$, are the algebraic anomalies that we wish to control.

The above analysis readily extends to higher–order single–input systems. In general, higher–order kernel estimates obtained with m–sequences have numerous spurious contributions analogous to the lower-order anomalies already described.

The situation only worsens for a multi–input nonlinear system. For example, consider a two-input system with nonzero kernels up to second order. In the standard approach (Sutter, 1987, 1992), a single m–sequence is the test signal for one input to this system and a long lag of this sequence is the second input. By an analysis exactly analogous to that above, it is straightforward to show that the estimate of the first–order kernels of this system will have contributions from all of the second–order kernels: two self-kernels and one cross-kernel. The estimates of these three second–order kernels will not only potentially contaminate each other but will also contain contributions from the first–order kernels for each input. Thus, for each second–order kernel, there are five kinds of anomalies, in addition to the $O(\frac{1}{M})$ parts.

### Elimination of Anomalies

Several approaches to the anomaly problem have been developed. In this section, we focus on the inverse–repeat (IR) method (Ream, 1970; Sutter, 1992) which separates even– and odd–order kernel estimates from each other. In conjunction with the hybrid m–sequence method we describe, its effectiveness is expanded.

The IR method relies on measuring the second–order system's response of Eq. (13) to the original m–sequence, $m(t)$, and to a second input given by $m(t)$ inverted in polarity: $(1 \rightarrow -1, -1 \rightarrow 1)$. We denote the response to $m(t)$ by $r_+(t)$, and the response to the inverted input by $r_-(t)$. First–order contributions will be inverted in response to the inverted sequence, while the second–order contributions will maintain the same sign in both cases. Thus, subtraction of the first–order kernel estimate made with inverted sequence from the estimate made with the original sequence and averaging will produce an estimate of the first–order kernel that is free of second–order (and all higher, even-order) contamination. That is,

$$\hat{h}_1^{IR}(l) = \frac{1}{2}\left(\langle r_+(t) \cdot m(t-l)\rangle - \langle r_-(t) \cdot m(t-l)\rangle\right) \tag{19}$$

Adding two second–order estimates from $r_+(t)$ and $r_-(t)$ and averaging will annihilate aberrant contributions from first–order kernels:

$$\hat{h}_2^{IR}(l_1, l_2) = \frac{1}{2}\left(\langle r_+(t) \cdot m(t-l_1) \cdot m(t-l_2)\rangle + \langle r_-(t) \cdot m(t-l_1) \cdot m(t-l_2)\rangle\right) \tag{20}$$

For the second–order system in Eq. (13), this procedure would produce the estimates:

$$\hat{h}_1^{1+}(l) = L_1(l) + O(\frac{1}{M}) + \sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1, k_2)\delta(l, F_1(k_1, k_2)) \tag{21}$$

$$\hat{h}_1^{1-}(l) = -L_1(l) + O(\frac{1}{M}) + \sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1, k_2)\delta(l, F_1(k_1, k_2)) \tag{22}$$

$$\hat{h}_1^{IR}(l) = L_1(l) + O(\frac{1}{M}) \tag{23}$$

$$\hat{h}_2^{1+}(l_1,l_2) = L_2(l_1,l_2) + O(\frac{1}{M}) + \sum_{k=0}^{T} L_1(k)\delta(k_1\, F_1(l_1,l_2)) +$$

$$\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)\delta(F_1(l_1,l_2),F_1(k_1,k_2)) \tag{24}$$

$$\hat{h}_2^{1-}(l_1,l_2) = L_2(l_1,l_2) + O(\frac{1}{M}) - \sum_{k=0}^{T} L_1(k)\delta(k_1\, F_1(l_1,l_2)) +$$

$$\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} l_2(k_1,k_2)\delta(F_1(l_1,L_2),F_1(k_1,k_2)) \tag{25}$$

$$\hat{h}_2^{IR(1)}(l_1,l_2) = L_2(l_1,l_2) + O(\frac{1}{M}) + \sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)\delta(F_1(l_1,l_2),F_1(k_1,k_2)) \tag{26}$$

where $\hat{}I\ R$ denotes the estimates derived from applying Eqs. (19) and (20), and $\hat{}1+$ and $\hat{}1-$ denote the estimates due to $m_1(t)$ and $m_1(t)$ inverted in polarity, respectively. The IR method thus separates even–order and odd–order kernel estimates from each other. However, it cannot resolve the problem of higher–order ($n \geq 2$) kernel estimates of the same parity (even or odd) from contaminating each other (see Eq. (26)).

Sutter (1992) has suggested the use of m–sequences with recursion terms that exceed the memory of the nonlinear system being tested to eliminate this kind of anomaly. For example, for a system with a memory of 15 time steps, an appropriate m–sequence would be of length $2^{15} - 1$. Although this guarantees (via the nature of $F$) clean separation of all kernels up to memory 15, this method becomes exceedingly onerous for multi–input nonlinear systems, or for a system whose memory is long. For a system with two inputs of memory 15, clean separation would require a single input of length $2^{30} - 1$! For physiological experiments, such long sequences would exceed the period during which the preparation is stable. One motivation for the hybrid method is to remove contamination without using exceedingly long sequences or requiring an intimate knowledge of the recursive structure of the stimulus.

## THE HYBRID M–SEQUENCE METHOD

In this section, we examine the benefits of presenting a sum of m–sequences as a signal to a single–input system. That is, our input is $s(t) = m_1(t) + m_2(t)$, where $m_1(t)$ and $m_2(t)$ are distinct m–sequences whose lengths are relatively prime. The new stimulus can have three values: (-2, 0, and 2). This stimulus requires a new algorithm for estimating the Wiener kernels, but the reader should easily see the similarities between the new algorithm and the Lee and Schetzen (1965) algorithm.

We begin by considering the system described by Eq. (2) truncated after kernels of second order. The response of this system to $s(t) = m_1(t) + m_2(t)$ is:

$$r_(t) = L_0 - \sum_{k=0}^{T} L_1(k)m_1(t-k) + \sum_{k=0}^{T} L_1(k)m_2(t-k)$$

$$+\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)m_1(t-k_1)m_1(t-k_2)+\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)m_2(t-k_1)m_2(t-k_2)$$

$$+2\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)\,m_1(t-k_1)m_2(t-k_2) \tag{27}$$

In Eq. (27), the system responds to each sequence individually and to pairwise combinations of sequences.

As before, to calculate the zeroth–order kernel, the response is averaged over an entire cycle of the combined signal.

$$\hat{h}_0 = \langle r(t)\rangle \tag{28}$$

The first–order kernel may be calculated by cross–correlating the response with either m- sequence in the input:

$$\hat{h}_1^1(l) = \langle r(t)\cdot m_1(t-l)\rangle \tag{29}$$

$$\hat{h}_1^2(l) = \langle r(t)\cdot m_2(t-l)\rangle \tag{30}$$

These are two independent estimates.

Estimates of the second–order kernel require that the lengths of $m_1(t)$ and $m_2(t)$ are relatively prime (i.e., they have no common factors apart from 1). As seen in Appendix II, this implies that

$$\langle m_1(t-l_1)\cdot m_2(t-l_2)\rangle = O(\frac{1}{M_1\cdot M_2}) \tag{31}$$

where $M_1$ and $M_2$ are the lengths of the $m_1(t)$ and $m_2(t)$ respectively. In general, for the set of m–sequences, $\{m_1(t), m_2(t), m_3(t), \ldots, m_n(t)\}$,

$$\langle m_1(t-l_1)\cdot m_2(t-l_2)\cdot m_3(t-l_3)\cdot \ldots m_n(t-l_n)\rangle = O(\frac{1}{M_1\cdot M_2\cdot \ldots M_n}) \tag{32}$$

if the lengths of each $m_i(t)$ are relatively prime to each other. Equation (31) is the basis of an estimation formula for the second–order kernel:

$$\hat{h}_2^{1,2}(l_1,l_2) = \frac{1}{2!}\langle r(t)\cdot m_1(t-l_1)\cdot m_2(t-l_2)\rangle \tag{33}$$

That is, to calculate the second–order kernel, the response is cross–correlated with both sequences in the input. Along the diagonal $l_1 = l_2$, the same cross–correlation as (33) can be used. The ability to measure the points on the diagonal of the kernel also distinguishes the hybrid method from the standard m–sequence method, to which these points are inaccessible.

We now see how this algorithm performs. Applying the cross–correlations of Eqs. (28)–(30) and (33) to the system of Eq. (2) truncated at second–order leads to:

$$\hat{h}_0 = L_0 + O(\frac{1}{M}) + 2\sum_{k=0}^{T} L_2(k,k) \tag{34}$$

$$\hat{h}_1^1(l) = L_1(l) + O(\frac{1}{M}) + \sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)\delta(l, F_1(k_1,k_2)) \tag{35}$$

$$\hat{h}_1^2(l) = L_1(l) + O(\frac{1}{M}) + \sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)\delta(l, F_2(k_1,k_2)) \tag{36}$$

$$\hat{h}_2^{1,2}(l_1,l_2) = L_2(l_1,l_2) + O(\frac{1}{M_1 \cdot M_2}) \tag{37}$$

The first–order kernels have second–order contributions, but these may be eliminated by an inverse repeat. The key point is that anomalies in the estimate of $\hat{h}_2^{1,2}(l_1,l_2)$ due to algebraic anomalies of the m–sequence have been completely eliminated (compare Eqs. (26) and (37)).

This analysis is somewhat artificial in that it is unlikely that a real system has a Volterra representation which is truncated after order two. Contributions from third–order Volterra terms will necessarily overlap on first–order estimates, even with this hybrid approach. However, comparing the two estimates, $\hat{h}_1^1(l)$ and $\hat{h}_1^2(l)$ gives a convenient experimental check on the presence of such "kernel noise" in the estimates of $h_1(l)$, because this "noise" depends on different mappings, $F_1$ and $F_2$.

### Higher–Order Systems

We have shown that the hybrid approach (a sum of m–sequences) can remove anomalous contributions which have the same parity as the kernel of interest. This is in contrast to the IR method, which can only remove anomalies on the basis of parity. This advantage of the hybrid approach extends to higher–order nonlinearities, provided that a longer input sequence is used. We consider the estimation of a third–order kernel via an input signal $s(t) = m_1(t) + m_2(t) + m_3(t)$, where the lengths of the three m–sequences are relatively prime in pairs (for example, 31, 63, and 127= 248,031). We will see that this provides for estimates of the third–order kernel to lag 31, free of anomalies due to the algebraic properties of the m–sequences. (The length of the stimulus $s(t)$, (31 x 63 x 127) = 248,031, though long, is nevertheless approximately 1/8000th as long as an m- sequence of order 31, which would be required by the use of a single m–sequence whose order exceeds the maximum lag of the kernel to be extracted).

The response of a third–order Volterra system to the input $s(t) = m_1(t) + m_2(t) + m_3(t)$ is:

$$r(t) = L_0 + \sum_{k=0}^{T} L_1(k)\left(\sum_{p=1}^{3} m_p(t-k)\right) +$$

$$\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)\left(\sum_{p,q=1}^{3} m_p(t-k_1)m_q(t-k_2)\right) +$$

$$\sum_{k_1=0}^{T}\sum_{k_2=0}^{T}\sum_{k_3=0}^{T} L_3(k_1,k_2,k_3)\left(\sum_{p,q,r=1}^{3} m_p(t-k_1)m_q(t-k_2)m_r(t-k_3)\right) \tag{38}$$

Applying Eq. (28), the estimated zeroth–order kernel is:

$$\hat{h}_0 = L_0 + O(\frac{1}{M}) + 3\sum_{k=0}^{T} L_2(k,k) \tag{39}$$

The first–order kernel can be estimated three ways by cross–correlating the response with any one of the m–sequences. The three estimates are ($p = 1,2,3$):

$$\hat{h}_p^1(l) = \langle r(t) \cdot m_p(t-l) \rangle \tag{40}$$

Substitution of Eq. (38) into Eq. (40) yields:

$$\hat{h}_p^1(l) = L_1(l) + O(\frac{1}{M}) + \sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2)\delta(F_p(k_1,k_2),l) +$$

$$\sum_{k_1=0}^{T}\sum_{k_2=0}^{T}\sum_{k_3=0}^{T} L_3(k_1,k_2,k_3)\delta(F_p(k_1,k_2,k_3),l) + 6\sum_{k=0}^{T} L_3(l,k,k) \tag{41}$$

where $F_p(k_1,k_2,k_3) = F_p(k_1,F_p(k_2,k_3))$. Note that the first–order kernel estimate has potential contributions from both the second- and third–order kernels. The last term is a contribution from the third–order kernel diagonal terms. This contribution is a proper part of the Wiener kernel. The second–order contribution can be removed by IR. The IR method does not remove the third–order, nondiagonal contribution since it is of the same parity (odd) as the kernel being estimated. The disparities between the three estimates $\hat{h}_1^1, \hat{h}_1^2$, and $\hat{h}_1^3$ can be used to gauge the significance of this higher–order overlap, since in each estimate, the third–order overlap depends on a different mapping $F$ (either $F_1$, $F_2$, or $F_3$).

With three m–sequences, it is possible to obtain three estimates of $h_2$. To estimate the second–order kernels, the response is cross–correlated with any two of the component m–sequences:

$$\hat{h}_2^{p,q}(l_1,l_2) = \frac{1}{2!}\langle r(t) \cdot (m_p(t-l_1) \cdot m_q(t-l_2)) \rangle (p \neq q) \tag{42}$$

Substitution of Eq. (38) into Eq. (42) yields:

$$\hat{h}_2^{p,q}(l_1,l_2) = L(l_1,l_2) + O(\frac{1}{M}) + \frac{3}{2}\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_3(l_2,k_1,k_2)\delta(F_p(k_1,k_2,),l_1) +$$

$$\frac{3}{2}\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_3(l_1,k_1,k_2)\delta(F_q(k_1,k_2,),l_2)) \tag{43}$$

Thus, there are three estimates for the second–order kernel, $\hat{h}_2^{1,2}, \hat{h}_2^{2,3}$, and $\hat{h}_2^{1,3}$. The third–order contamination of the second–order kernel can be estimated by comparing the quantities, or can be removed by IR, e.g.,

$$\hat{h}_2^{IR(1,2)} = \frac{1}{2}(\hat{h}_2^{1+2+} + \hat{h}_2^{1-2-}) \tag{44}$$

For the third–order kernel, the estimate is evaluated by cross-correlating the response with three m–sequences:

$$\hat{h}_3^{1,2,3}(l_1, l_2, l_3) = \frac{1}{3!}\langle r(t) \cdot m_1(t - l_1) \cdot m_2(t - l_2) \cdot m_3(t - l_3)\rangle \tag{45}$$

Substitution of Eq. (38) into Eq. (45) yields

$$\hat{h}_3^{1,2,3}(l_1, l_2, l_3) = L_3(l_1, l_2, l_3) + O(\frac{1}{M_1 \cdot M_2 \cdot M_3}). \tag{46}$$

The general form of the hybrid algorithm may now be stated: The input signal is a sum of $n$ distinct m–sequences of relatively prime lengths. The $n$th–order kernel is estimated by cross-correlating the response with the product of these $n$ m–sequences. The $k$th–order kernel ($k < n$) has $\frac{n!}{(n-k)!k!}$ estimates, each obtained by cross–correlation with a product of $k$ of the input m–sequences. By comparing these estimates, one can judge whether overlaps are significant and whether to average these estimates or combine them with IR measurements. Formally, for an input $s(t)$ consisting of a sum of $n$ m–sequence inputs, all of relatively prime length to the others ($s(t) = m_1(t) + m_2(t) + m_3(t) + ... + m_n(t)$), the kernel calculations are made as follows:

$$\hat{h}_0 = \langle r(t)\rangle \tag{47}$$

$$\hat{h}_1^p(l) = \langle r(t) \cdot m_p(t - l)\rangle \tag{48}$$

$$\hat{h}_2^{p,q}(l_1, l_2) = \frac{1}{2!}\langle r(t) \cdot m_p(t - l_1) \cdot m_q(t - l_2)\rangle \tag{49}$$

$$\vdots$$

$$\hat{h}_n^{1,2,\ldots,n}(l_1, l_2, \ldots, l_n) = \frac{1}{n!}\langle r(t) \cdot m_1(t - l_1) \cdot m_2(t - l_2) \ldots \cdot m_n(t - l_n)\rangle \tag{50}$$

The FMT can be used to calculate each one of these cross–correlations in exactly the same way that the FFT can be used to calculate the Fourier transform in more than one dimension (see below).

### Multi–Input Systems

The final step in the development of this method is to apply it to the identification of a multi–input nonlinear system. This extension is straightforward now that the foundations have been laid. To measure the kernel of a single-input system above, it is necessary that each m–sequence is longer than the system's memory. To extend this technique to a multi–input nonlinear system, each sequence must have a length which exceeds the system's memory multiplied by the number of inputs. Then each

input is assigned a tap, or lag, which is longer than the system's memory. The cross–correlation analysis proceeds exactly as before, and the kernels for each input are represented in the resulting cross–correlation shifted by an amount corresponding to its associated input lag.

Consider, for example, a two–input, second–order system. There are two first–order kernels, one for each input. There are three distinct second–order kernels; one involving the product of two values of the first input, one involving the product of two values of the second input, and a third that involves the product of the first and second inputs together. To resolve the second–order kernels, a sum of two relatively prime m–sequences is chosen for the input. Each individual sequence is of length at least $2T$ (where $T$ as before is the system's memory in discretized time units). The signal applied to the first input is $s_{[1]}(t) = m_1(t) + m_2(t)$, i.e., the sum of two m–sequences. The signal applied to the second input is $s_{[2]}(t) = m_1(t+T_1) + m_2(t+T_2)$, where $T_1$ and $T_2$ are lags which are greater than the memory of the system. These lags separate the contributions of the two inputs. The cross–correlation formulae, Eqs. (48) – (50), yield estimates of the zeroth–, first– and second–order kernels, except for the combinatorial factor on the left–hand side which must be adjusted for the second–order cross-kernel. Instead of $\frac{1}{2!}$, the coefficient is simply 1. For the $n$th–order multi–input kernel with $k$ different inputs, this coefficient is $\frac{k!(n-k)!}{n!}$.

The two–dimensional cross–correlation analysis (facilitated by the FMT) that yields estimates of the second–order kernels is graphically represented in Fig. 1. The response of the system is recorded and used to fill a two–dimensional array, the rows and columns of which are indexed by the position of the signal with respect to each of the two m–sequences at the time of the response. This array is transformed in one dimension by the FMT corresponding to the sequence, $m_1(t)$, and in the other dimension according to $m_2(t)$. The second–order kernels are separated by the lags $T_1$ and $T_2$ in the transformed array.

For higher–order, multi–input, nonlinear systems, the analysis proceeds the same way. For an $n$th–order system with $k$ inputs and memory $T$, $n$ lagged m–sequences are summed and applied to each of the $k$ inputs. Lags assigned to different inputs must differ by at least an amount $T$. Each sequence, therefore, is of length at least $kT$, and the lengths of the sequences, as above, must be relatively prime.

### Relationship to Wiener Kernels

Above we have discussed the measurement of system kernels with a stimulus made up of a sum of relatively prime–length m–sequences. In the original Wiener framework, the kernels form an orthogonal set of functionals with respect to Gaussian white–noise of a particular power. The measured kernels correspond then to orthogonal functionals with respect to this Gaussian white–noise input. The orthogonality assures that truncating this series at any given order minimizes the output prediction error in a least–squares sense.

The hybrid m–sequence method, on the other hand, uses a stimulus that is neither white nor strictly Gaussian. However, biological systems generally have a finite bandwidth, and stimulus frequencies that exceed this bandwidth will not influence the response. The highest frequency in the stimulus is determined by the length of the time interval between steps in the m–sequence. The shorter this interval the wider the bandwidth of the signal. The lowest frequency in the stimulus is determined by the length of the m–sequence since the stimulus repeats after this interval. As long as the these two parameters are set to span the frequency range of the

system under study, the frequencies present in the stimulus are a good approximation to those present in Gaussian white–noise.

In the above analysis, we have noted where diagonal elements have contributed to lower order kernels. These contributions depend on the power of the stimulus. Thus the Wiener kernels and the kernels estimated by the hybrid m–sequence are dependent on the power level of the stimulus used to estimate the kernels. For example, we noted above that the diagonal elements of the second–order kernel, $h_2$, contribute to $h_0$. If the stimulus is made up of two m–sequences of unit amplitude then this contribution is $2\sum_{k=0}^{T} h_2(k,k)$ (Eq. (34)). For a Gaussian white-noise stimulus of power, $P$, the corresponding contribution is $P\sum_{k=0}^{T} h_2(k,k)$. Thus the zeroth–order kernel measured with the sum of two m–sequences is an estimate of the Wiener kernel of equal power. In general, Wiener kernel estimates derived from a sum of $n$ m–sequences are matched for a Gaussian white-noise of power $n$. As the number of m–sequences in the stimulus is increased, the number of kernels that can be estimated grows. This is because the statistics of the stimulus become more and more like those of a Gaussian distribution, i.e., each value of the stimulus is a sum of independent random variables, and thus the Central Limit Theorem applies (Feller, 1968).

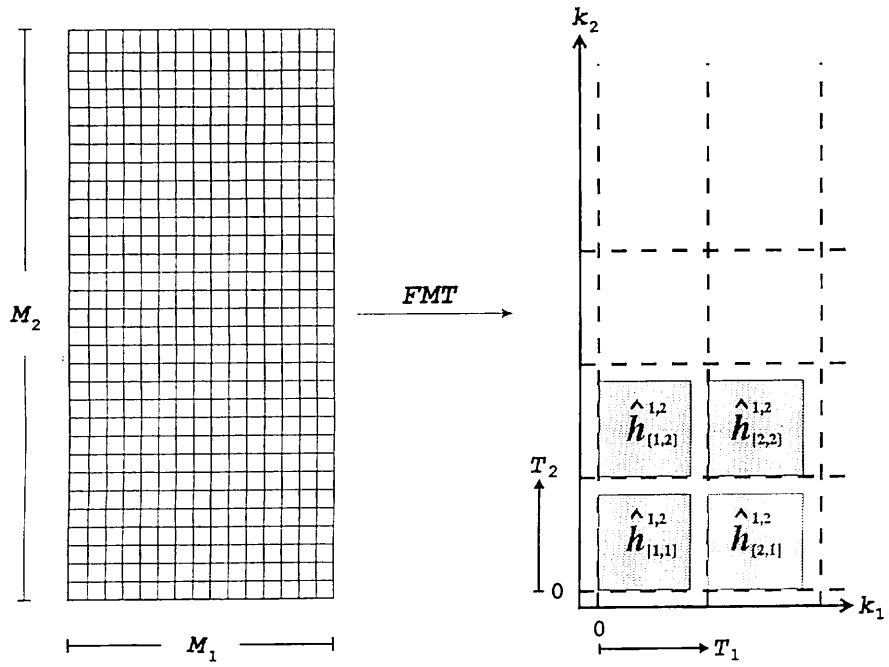


Figure 1. This figure shows the scheme for estimating the Wiener kernels using the FMT and the sum–of–m–sequences signal. The response of the multi–input (in this case, two–input) system to the sum signal is used to fill an array indexed by the position of the signal with respect to its component m–sequences. For two m–sequences, this array is two dimensional. For the experiment diagrammed, two m–sequences are summed in each of two signals, $s_{[1]}(t)$ and $s_{[2]}(t)$. Both signals are the sum of $m_1(t)$ and $m_2(t)$ but $s_{[1]}(t) = m_1(t) + m_2(t)$, and $s_{[2]}(t) = m_1(t+T_1) + m_2(t+T_2)$. The kernels are then calculated by transforming the array along both dimensions with the FMT. The transformed space contains the second–order kernels $\hat{h}^{1,2}_{[1,1]}$, $\hat{h}^{1,2}_{[2,2]}$, $\hat{h}^{1,2}_{[:,2]}$, and $\hat{h}^{1,2}_{[2,1]}$ where the subscripts indicate the signals that are acted on by the kernels, and the superscripts indicate that both m–sequences are used in the kernel estimates. The cross–kernels, $\hat{h}^{1,2}_{[1,2]}$, and $\hat{h}^{1,2}_{[2,1]}$, are two independent estimates of the same kernel with the time axes interchanged (hence, the order of the signals is reversed in the subscript).

Thus stimulus design depends in part on *a priori* knowledge of the system one is trying to study. The choice of time interval and m–sequence length require knowledge of the frequency response and memory of the system. Furthermore, it is helpful to know approximately the highest-order interaction likely to be present in the system's response. A hybrid routine involving $k$ m–sequence inputs can only rigorously measure kernels up to order $k$. Components of higher–order in the response will confound the calculation of lower–order kernels. Indeed, one must always bear in mind that "noise" in a kernel measurement may represent contamination of the measurement by some kernel whose order exceeds the capacity of the analytical technique.

### Relationship to the Sum–of–Sinusoids Approach

In the sum-of-sinusoids method, separation of nonlinear contributions of different orders is achieved by choosing a set of input frequencies which are nearly incommensurate. This provides an input signal which densely samples the input phase space. The condition of near-incommensurateness makes the component sinusoids nearly independent, and thus guarantees orthogonality of kernel estimates of different orders. The hybrid m–sequence method likewise uses a sum of m–sequences of relatively prime lengths. The component m–sequences can be thought of as the time–domain analogues of the nearly incommensurate frequencies. They sample the time–lag space densely and independently, and thus allow calculation of Wiener kernels with little kernel cross-contamination. Although the frequency approach promises better kernel separation, the hybrid m–sequence method offers a simpler routine for multi–input nonlinear system identification, since shifts of the same m–sequences that are used as the signal to one input can be used as the signal to another input.

## PHYSIOLOGICAL RESULTS

The motivation for developing this procedure was to investigate multi–input nonlinear processes in the primate visual system. We describe here application of this procedure to the primate retinal ganglion cell (RGC) (Benardete et al., 1992). The RGCs form the output units of the retina. The RGCs that project to the parvocellular layers of the LGN (P cells) represent an early processing channel for luminance, color, and spatial interactions. The method described above has allowed us to examine nonlinear, pairwise interactions between color and luminance and the surround and center in the receptive field of P cells. The first–order responses obtained simultaneously with these inputs allowed us to relate our measurements to previous work which was limited to linear and quasi–linear analysis. We describe here one of the experiments we have done to address linear and nonlinear spatial interactions in the P cell. The visual stimulus consisted of a spot and an annulus produced on a CRT monitor and centered on the receptive field of the cell. The contrast of the spot and the annulus was modulated by a sum of two m–sequences of relatively prime length. The sequences in the signal used to modulate the annulus were long lags of the sequences used to modulate the spot. The lengths and the step interval of the two m–sequences were chosen to span the temporal frequency range of the RGC. The m–sequences were of length 31 and 63, and the time step was 14.8 ms. For the annulus signal, the sequences of length 31 and 63 were lagged by 16 and 32, respectively. The output variable, $r(t)$, was the firing rate of the neuron recorded as

extracellular potentials in the LGN of the primate (*Macaca fascicularis*) (Kaplan and Shapley, 1986).

For kernel extraction, the IR method was used in conjunction with the hybrid signal. The second–order kernels represent the response of the system to pairwise combinations of center and/or surround contrast signals. The estimates from three presentations based on different m–sequences of the same lengths were averaged. Figure 2 displays these kernel estimates. The first–order kernels for the center and surround receptive field mechanisms of the P cell (Figs. 2A and 2B) show that the

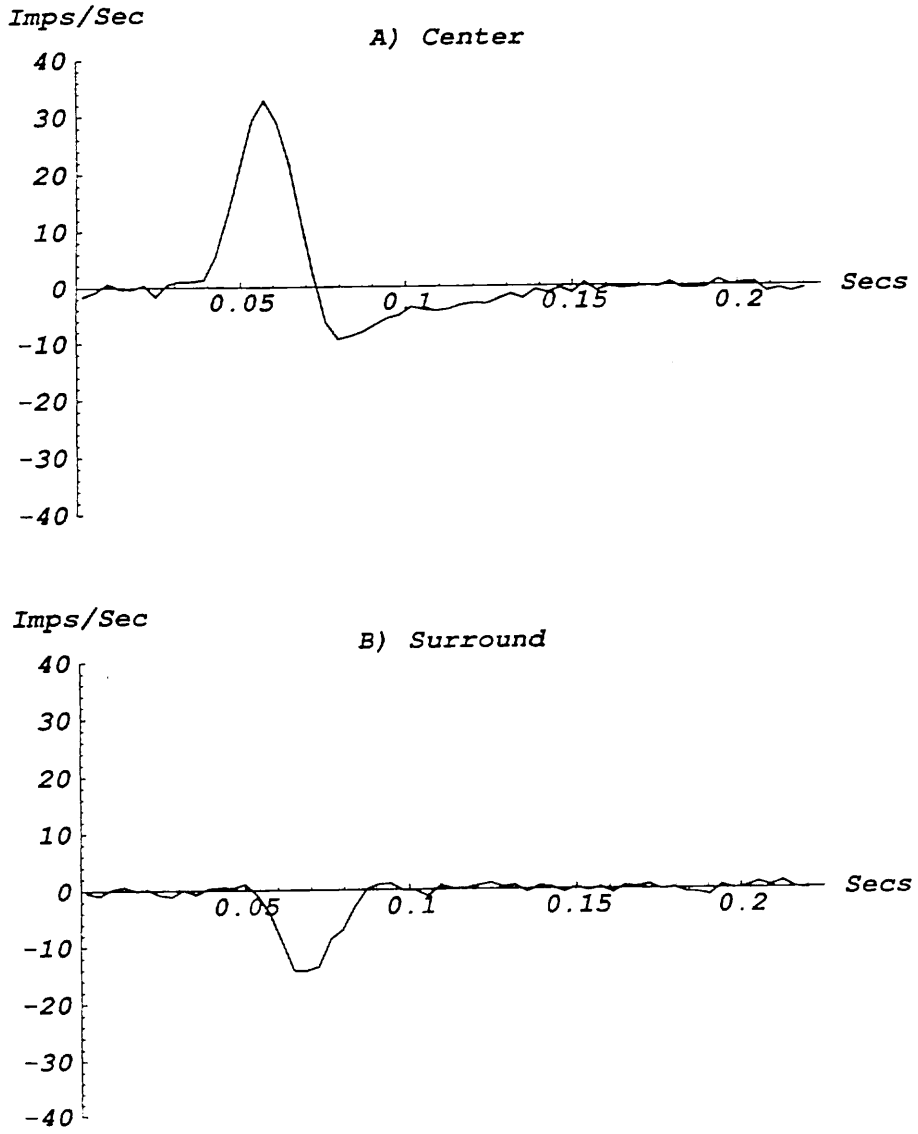


Figure 2 A) The first–order Wiener kernel estimate for a P ON cell in response to a small contrast–modulated spot (0.25 deg) placed over the center of the receptive field. B) The first–order Wiener kernel estimate in response to a contrast–modulated annulus (inner diameter: 0.45 deg; outer diameter: 3.5 deg) in the far surround of the receptive field of the same cell presented simultaneously with the spot. The two first–order kernels are estimated from the same response to the hybrid stimulus. Note that the response of the surround is of opposite polarity to that of the center and is of greater latency.

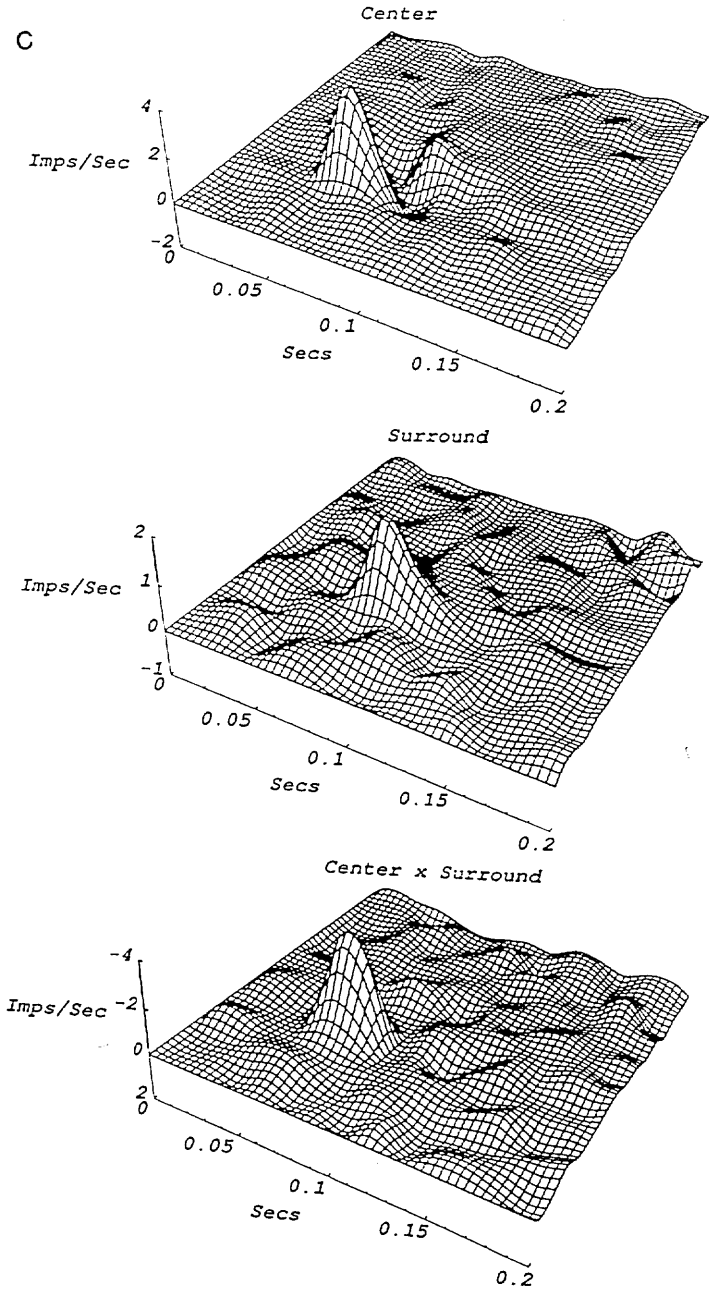


Figure 2 C) Surface plots of the second–order Wiener kernels for the center (spot x spot), surround (annulus x annulus), and center x surround (spot x annulus) interaction calculated with the hybrid m–sequence method. A cubic spline procedure was performed on the estimated kernel values to fill in the surface. The x and y axes in the surface plots indicate the time from the origin for the kernel values. For the center and surround ("self–") kernels, both axes indicate time for the center and surround signals, respectively (i.e., the kernels are symmetric). For the center x surround ("cross–") kernel, the near axis shows time from the origin for the center signal. The time axis for the surround signal is hidden by the surface. Cross–kernels are not, in general, symmetric, as illustrated by the center x surround kernel of this P cell. The presence of these second–order kernels demonstrates nonlinear interactions in the center and surround mechanisms of P cells that have not previously been identified.

surround response is more delayed and less biphasic than the center response. The second–order kernels (Fig. 2C) for the same P cell show that both the center and surround of the P cell produce nonlinear responses. In addition, the center–surround cross–kernel shows that the center and surround of the P cell interact in a multiplicative way. This type of interaction is a new finding not addressed by standard linear and quasi–linear models (Ingling and Martinez–Uriegas, 1983; Derrington et al., 1984; Kaplan et al., 1990). Simulations of these data show that these results cannot be accounted for on the basis of some simple physiological nonlinearity such as response saturation or truncation. Currently, we are using these data to develop a nonlinear model of the P cell receptive field.

## DISCUSSION

The theoretical framework of Wiener nonlinear systems analysis provides an attractive and precise formalism for categorizing and modeling any system. Direct application of this approach is impossible because true Gaussian white–noise is an ideal that cannot be realized in practice. This limitation has led to many pioneering efforts in designing more practical system identification algorithms that retain the theoretical advantages of the Wiener approach.

Deterministic signals offer certain advantages in this context, since the experimenter knows in advance the statistical deviations of the signal from a Gaussian white–noise. Within the deterministic framework, one of the most promising approaches is the m–sequence method, as developed by Sutter (1987, 1992). However, the algebraic structure of the m–sequences leads to anomalies in kernel estimates for nonlinear systems, especially for those with many inputs. In this paper, we show how many of these anomalies can be eliminated by a method analogous to the sum-of-sinusoids method (Victor and Shapley, 1980). By choosing a sum of m–sequences of relatively prime lengths, the cross–correlation properties of the stimulus were refined to enable a simple and accurate estimation of multi–input nonlinear kernels.

The goal in designing a useful Wiener-like system identification procedure is to separate the system into kernels of various orders. Reduction of the cross-contamination of the calculated kernels can be achieved by appropriate stimulus design, but since the stability of physiological preparations is limited, the recording times are also best kept as short as possible. The sum–of–m–sequences is economical in this regard as well, since it samples the pertinent region of stimulus space densely and uniformly.

## APPENDIX I

This appendix is a brief introduction to generalized Wiener kernels from the point of view of Victor and Knight (1979). A Volterra series defines a physical system in terms of a hierarchy of kernels, each of which can be thought of as a transducer, $\mu_j$, of a particular order $j$. The order refers to the number of time lags of the stimulus that are multiplied and weighted by the transducer.

The transducers, $\mu_j$, are members of a vector space, $M_j$, of transducers of Volterra order $j$. Basis functions for this vector space are:

$$D_{j,\vec{\tau}}(s)(t) = \Pi_{k=1}^{j} s(t - \tau_k) \qquad \text{(A1.1)}$$

which take the product of $j$ time lags, $\tau_k$, of the stimulus, $s(t)$. For a single–input system, the ordering of the $s(t-\tau_k)$ is irrelevant so that the $D_{j,\vec{\tau}}$ are symmetric functions of their arguments. For stimuli, $s(t)$, whose value at each time is independently chosen, they form a set of linearly independent basis elements only as long as each of the time lags is different.

The vector spaces, $M_j$, are not orthogonal. However, each $M_j$ can be projected onto a new set of orthogonal vector spaces, $K_j$, in which each basis function has lower order correction terms. Each transducer in the $K_j$ space is orthogonal to any transducer in a subspace $K_l$ of a different order.

The Wiener characterization of an unknown transducer is the projection of that transducer onto the vector spaces, $K_j$. The projection of a transducer $\mu_j$ in $M_j$ into its corresponding orthogonal space $K_j$ can be written:

$$\mu_j = \sum_{k=0}^{j} \mu_j^{(k)} \tag{A1.2}$$

as sum of a $j$th–order term, $\mu_j^{(j)}$ , and lower order correction terms, $\mu_j^{(k)}, 0 \leq k < j$. These lower-order correction terms depend on the power in the stimulus. They differentiate the kernels of the Wiener orthogonal functional expansion from those of the Volterra series.

## APPENDIX II

The goal of this Appendix is to show that a sum of m–sequences is an especially useful signal for nonlinear systems analysis, if the lengths of the component m-sequences are **relatively prime** (i.e., their lengths are numbers that share no common factor except for 1). We show that the $n$th–order Wiener kernel of a system can be estimated by cross-correlating the response to this test signal with the product of its $n$ component m–sequences.

We require that each component m–sequence has a length greater than the memory of the system, $T$. We denote the length of each component signal, $m_i(t)$, by $M_i$. The relative primality condition implies that the length of the sum signal, $M^*$, is given by $M_1 \cdot M_2 \ldots M_n$.

Consider the response of a second–order system to a sum of two m–sequences, $m_1(t) + m_2(t)$:

$$r(t) = L_0 + \sum_{k=0}^{T} L_1(k) m_1(t-k) + \sum_{k=0}^{T} L_1(k) m_2(t-k) +$$

$$\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2) m_1(t-k_1) m_1(t-k_2) + \sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2) m_2(t-k_1) m_2(t-k_2) +$$

$$2\sum_{k_1=0}^{T}\sum_{k_2=0}^{T} L_2(k_1,k_2) m_1(t-k_1) m_2(t-k_2). \tag{A2.1}$$

To estimate the second–order Wiener kernel, $h_2(l_1,l_2)$, the following cross–correlation:

$$\hat{h}_2^{1,2}(l_1,l_2) = \frac{1}{2!}\langle r(t) \cdot m_1(t-l_1) \cdot m_2(t-l_2)\rangle \tag{A2.2}$$

is valid, as long as no other points of $L_2$ contribute to the estimate (see Eq. 18). In order for this to be true, products of shifts of $m_1(t)$ and $m_2(t)$ must be uncorrelated with any other shifts of $m_1(t)$ and $m_2(t)$. We will show that this is the case:

$$\langle m_1(t-l_1)\cdot m_2(t-l_2)\cdot m_1(t-k_1)\cdot m_2(t-k_2)\rangle = O(\frac{1}{M^*}) \text{ if } l_1 \neq k_1, l_2 \neq k_2 \quad \text{(A2.3)}$$

where $\langle\rangle$ denotes an average over the whole sum–of–m–sequences signal. From Eq. (6), the product of two shifts of the same m–sequence is another shift of the same m–sequence. Thus, Eq. (A2.3) is equivalent to:

$$\langle m_1(t-q_1)\cdot m_2(t-q_2)\rangle = O(\frac{1}{M^*}) \quad \text{(A2.4)}$$

where $q_1 = F_1(l_1, k_1)$ and $q_2 = F_2(l_2, k_2)$. Because of the relative-primality condition (see below) each element of $m_1(t)$ occurs once and only once with each element of $m_2(t)$ in composite sequence of length $M^* = M_1 \cdot M_2$. That is, the values of the m–sequences are independent over the cycle length, and the average in Eq. (A2.4) separates:

$$\langle m_1(t-q_1)\cdot m_2(t-q_2)\rangle = \langle m_1(t-q_1)\rangle_1 \cdot \langle m_2(t-q_2)\rangle_2 = \frac{1}{M_1 \cdot M_2} \quad \text{(A2.5)}$$

where $\langle\rangle_1$ indicates an average over $m_1(t)$ and $\langle\rangle_2$ indicates an average over $m_2(t)$. Note that if $l_1 = k_1$ but $l_2 \neq k_2$, then the correlation of Eq. (A2.3) is still small but it is $O(\frac{1}{M_2})$ since one term in (A2.5) is eliminated.

The fact that every element of $m_1(t)$ appears once and only once with every element of $m_2(t)$ in a cycle of length $M_1 \cdot M_2$ can be seen as follows: Let $x$ represent the position of an element in the sum–of–m–sequences signal. That is, for the first element of the sum signal, $x = 0$; for the second, $x = 1$, and so on. Let $a$ and $b$ represent the corresponding quantity for $m_1(t)$ (length $M_1$) and $m_2(t)$ (length $M_2$) respectively. Then, there is a unique $x$ in the range from 0 to ($M_1 \cdot M_2$ - 1) that solves the congruence equations:

$$x \equiv a(mod M_1) \quad \text{(A2.6)}$$

$$x \equiv b(mod M_2) \quad \text{(A2.7)}$$

if and only if $M_1$ and $M_2$ are relatively prime. This is a restatement of the Chinese Remainder Theorem of elementary number theory (Pinter, 1990). Fig. (A2.1) illustrates how the sum–of–m–sequences signal presents every element of one component m–sequence in conjunction with every element of the other component m–sequence once and only once. The sum signal, as shown in Fig. (A2.1), traces out a path which samples the plane of pairwise values evenly and exhaustively.

For a signal which is the sum of $n$ m–sequences, these ideas generalize to a system of $n$ congruence equations, provided that $M_1, M_2, \ldots, M_n$ are relatively prime. This leads to:

$$\langle m_1(t-q_1)\cdot m_2(t-q_2)\cdot \ldots \cdot m_n(t-q_n)\rangle = O(\frac{1}{M^*}) \quad \text{(A2.8)}$$

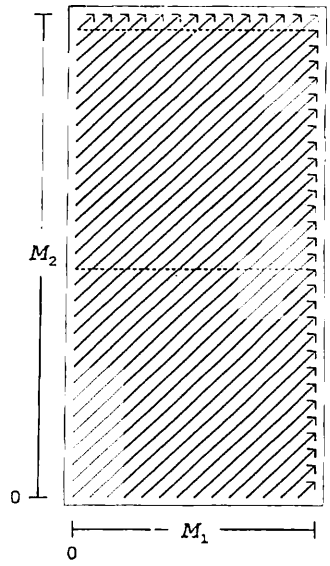


Figure A2.1. A geometric rendering of the sum–of–m–sequences signal for the sum of two m–sequences of relatively prime length. The horizontal and vertical axes indicate the position of the elements of the sum signal in the sequences $m_1(t)$ (length $M_1$) and $m_2(t)$ (length $M_2$), respectively. The sum signal starts out in the bottom left hand corner (0, 0) and continues to the far right and then wraps around to (0, $M_1$) (dashed line). As time progresses, the signal continues to move up and to the right, wrap around, and so on. In this way, the entire space is filled by the sum signal of length $M_1 \cdot M_2$. Each element of $m_1(t)$ appears once and only once with each element of $m_2(t)$ in the signal cycle. This structure provides statistical independence of the two component signals and improves Wiener kernel estimates.

Consequently,

$$\langle m_1(t-l_1)\cdot m_2(t-l_2)\ldots m_n(t-l_n)\cdot m_1(t-k_1)\cdot m_2(t-k_2)\ldots$$
$$m_n(t-k_n)\rangle = O(\frac{1}{M^*}) \qquad \text{(A2.9)}$$

if $l_1 \neq k_1, \ldots, l_n \neq k_n$. As in the second–order case discussed above, if some but not all of the $l_i$'s and $k_i$'s are equal, the cross–correlation Eq. (A2.9) is still small. Eq. (A2.2) generalizes to provide an estimate of the $n$th-order Wiener kernel measured with a sum–of– $n$–m–sequences signal:

$$\hat{h}_n^{1,2,\ldots,n}(l_1,l_2,\ldots,l_n) = \frac{1}{n!}\langle r(t)\cdot m_1(t-l_1)\cdot m_2(t-l_2)\cdot\ldots\cdot m_n(t-l_n)\rangle \qquad \text{(A2.10)}$$

Eq. (A2.9) guarantees near elimination of overlaps in this estimate. The main point is that the relatively prime lengths of the component m–sequences allow for averages over the entire stimulus cycle to be broken down into independent averages over each component m–sequence.

## Acknowlegements

We thank Dr. Ehud Kaplan for his experimental assistance and suggestions. EB is supported in part by NIH EY4888 and NIH EY1428. JV is supported in part by NIH EY9314 and NIH EY7977.

## REFERENCES

1. Barker, H.A., and Pradisthayon, T. (1970) High–order autocorrelation functions of pseudoramdom signals based on *M*–sequences. *Proc. IEE*, **177**:1857–1863.
2. Benardete, E.A., Victor, J.D. and Kaplan, E. (1992) Temporal properties of primate P retinal ganglion cells investigated with a new discrete multi–level stimulus. *Invest. Opthalmol. & Visual Sci.*, 33, abstr. #3591.
3. Derrington, A.M., Krauskopf, J. and Lennie, P. (1984) Chromatic mechanisms in lateral geniculate nucleus of macaque. *J. Physiol. (London)*, 357:241-265.
4. Feller, W. (1968) *An Introduction to Probability Theory and Statistics*, 3rd ed., John Wiley & Sons, New York, New York.
5. Golomb, S.W. (1968) *Shift Register Sequences*, Holden–Day, San Francisco, California.
6. Gyftopoulos, E.P., and Hooper, R.J. (1964) Signals for transfer function measurement in nonlinear systems. *Noise Analysis in Nuclear Systems*, USAEC Symposium series 4, TID–7679.
7. Ingling, C.R. and Martinez-Uriegas, E. (1983) The relationship between spectral sensitivity and spatial sensitivity in the primate r–g X–channel. *Vision Res.*, 23:1495-1500.
8. Kaplan, E. and Shapley, R.M. (1986) The primate retina contains two types of ganglion cells with high and low contrast sensitivity. *PNAS USA*, 83:2755-2757.
9. Kaplan, E., Lee, B.B. and Shapley, R.M. (1990) New views of primate retinal function, In: *Progress in Retinal Research, Vol. 9*, N.N. Osborne & J.G. Chader, eds., Pergamon Press, New York, New York.
10. Klein, S. (1987) Relationships between kernels measured with different stimuli, In: *Advanced Methods of Physiological Systems Modeling, Volume I*, V.Z. Marmarelis, (Ed.), Biomedical Simulations Resource, University of Southern California, Los Angeles, California.
11. Klein, S.A. (1992) Optimizing the estimation of nonlinear kernels, In: *Nonlinear Vision: Determination of Neural Receptive Fields, Function, and Networks*, R.B. Pinter & B. Nabet, Eds., CRC Press, Boca Raton, Florida.
12. Lee, Y.N. and Schetzen, M. (1965) Measurement of the kernels of a nonlinear system by cross–correlation. *Int. J. Control*, 2:237-254.
13. Marmarelis, P.Z. and Naka, K.-I. (1972) White–noise analysis of a neuron chain: An application of the Wiener theory. *Science*, 175:1276-1278.
14. Marmarelis, P.Z. and Marmarelis, V.Z. (1978) *Analysis of Physiological Systems: The White–Noise Approach*, Plenum Press, New York, New York.
15. Pinter, C.C. (1990) *A Book of Abstract Algebra*, McGraw-Hill, New York, New York.
16. Ream, N. (1970) Nonlinear identification using inverse–repeat *m*–sequences. *Proc. IEE*, **117**: 213–218.
17. Schetzen, M. (1980) *The Volterra and Wiener Theories of Nonlinear Systems*, John Wiley & Sons, New York, New York.
18. Sutter, E.E. (1987) A practical nonstochastic approach to nonlinear time–domain analysis, In: *Advanced Methods of Physiological Systems Modeling, Vol. I*, V.Z. Marmarelis, ed., Biomedical Simulations Resource, University of Southern California, Los Angeles, California.
19. Sutter, E.E. (1991) The fast m–transform: a fast computation of cross–correlation with binary m–sequences. *SIAM J. Comput.*, 20(4):686-694.
20. Sutter, E.E. (1992) A deterministic approach to nonlinear systems analysis, In: *Nonlinear Vision: Determination of Neural Receptive Fields, Function, and Networks*, R.B. Pinter & B. Nabet, Eds., CRC Press, Boca Raton, Florida.
21. Victor, J.D. and Knight, B.W. (1979) Nonlinear analysis with an arbitrary stimulus ensemble. *Q. Appl. Math.*, 37:113-136.
22. Victor, J.D. and Shapley, R.M. (1979) The nonlinear pathway of Y ganglion cells in the cat retina. *J. Gen. Physiol.*, 74:671-689.

23. Victor, J.D. and Shapley, R.M. (1980) A method of nonlinear analysis in the frequency domain. *Biophys. J.*, **29**:459-484.

24. Victor, J.D. (1991) Asymptotic approach of generalized functional expansions to Wiener kernels. *Ann. Biomed. Eng.*, **19**:383-399.

25. Victor, J.D. (1992) Nonlinear systems analysis in vision: Overview of kernel methods, In: *Nonlinear Vision: Determination of Neural Receptive Fields, Function, and Networks*, R.B. Pinter & B. Nabet, Eds., CRC Press, Boca Raton, Florida.

26. Volterra, V. (1932) *The Theory of Functionals and of Integral and Integro–Differential Equations*, Blackie, London, England.

27. Wiener, N. (1958) *Nonlinear Problems in Random Theory*, John Wiley & Sons, New York, New York.